

AD-A045 627

GENERAL ELECTRIC CO CINCINNATI OHIO AIRCRAFT ENGINE GROUP F/G 21/5
DEVELOPMENT OF EMISSIONS MEASUREMENT TECHNIQUES FOR AFTERBURNIN--ETC(U)
OCT 75 W C COLLEY, D R FERGUSON, M A SMITH F33615-73-C-2047
R75AE6459 AFAPL-TR-75-52-SUPPL-2 NL

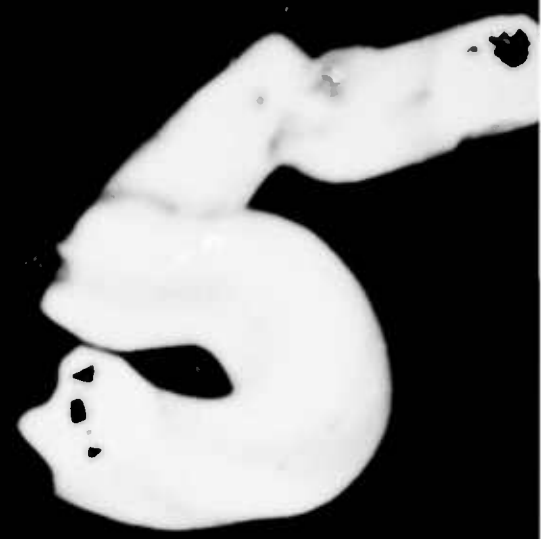
UNCLASSIFIED

1 of 5
ADA045 627

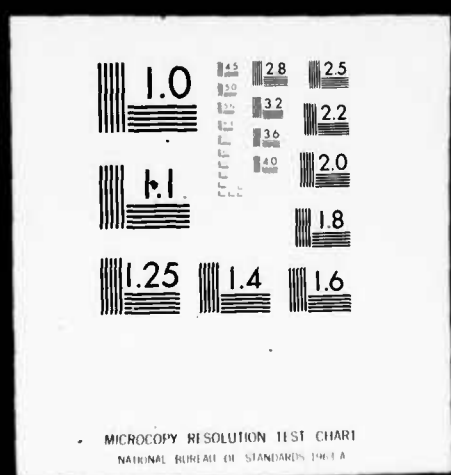


1

OF



ADA045 627



AFAPL-TR-75-52, Supplement 2

R75AEG459

AD45491
A019094

①
B.S.

DEVELOPMENT OF EMISSIONS MEASUREMENT TECHNIQUES FOR AFTERBURNING TURBINE ENGINES

Supplement 2 - Afterburner Plume Computer Program
User's Manual

W.C. Colley
D.R. Ferguson
M.A. Smith

GENERAL ELECTRIC COMPANY
Aircraft Engine Group
Cincinnati, Ohio 45215

October 1975

D D C
RECEIVED
OCT 14 1977
ALLEN

Q

Technical Report AFAPL-TR-75-52, Supplement 2

Approved for public release;
distribution unlimited

Prepared for

AIR FORCE AERO PROPULSION LABORATORY
Air Force Wright Aeronautical Laboratories
Wright-Patterson Air Force Base, Ohio 45433

AD A045627

AD NO. _____
DDC FILE COPY

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

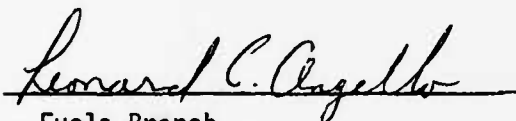
Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

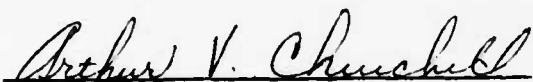
This report has been reviewed by the Information Office (OI) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

This technical report has been reviewed and is approved for publication.



Fuels Branch
Fuels and Lubrication Division

FOR THE COMMANDER



Chief, Fuels Branch
Fuels and Lubrication Division

"If your address has changed, if you wish to be removed from our mailing list, or if the addressee is no longer employed by your organization please notify AFAPL/SFF, W-PAFB, OH 45433 to help us maintain a current mailing list".

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER 18 AFAPL-TR-75-52, Supplement 2 ✓	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER 9 Technical rept.
4. TITLE (and Subtitle) DEVELOPMENT OF EMISSIONS MEASUREMENT TECHNIQUES FOR AFTERBURNING TURBINE ENGINES. SUPPLEMENT 2. AFTERBURNER PLUME COMPUTER PROGRAM USER'S MANUAL.	5. TYPE OF REPORT & PERIOD COVERED	6. PERFORMING ORG. REPORT NUMBER 14 R75AEG459
7. AUTHOR(s) 10 W.C./Colley, D.R./Ferguson and M.A./Smith	8. CONTRACT OR GRANT NUMBER(s) 15 F33615-73-C-2047 ✓	9. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 16 Project 1900
10. PERFORMING ORGANIZATION NAME AND ADDRESS General Electric Company Aircraft Engine Group Cincinnati, Ohio 45215	11. CONTROLLING OFFICE NAME AND ADDRESS Air Force Aero Propulsion Laboratory (AFAPL/SFF) Air Force Wright Aeronautical Laboratories Wright-Patterson AFB, Ohio 45433	12. REPORT DATE 17 Oct 1975 75
13. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	14. NUMBER OF PAGES 454 12 461 p.	15. SECURITY CLASS. (of this Report) Unclassified
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Afterburner Emissions Exhaust Plume Turbine Engine Environmental Pollution Exhaust Plume Model Emissions Exhaust Chemical Kinetics Plume Mixing		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) Computer program "PLUMOD" description and users' manual. PLUMOD is an integral part of the exhaust emissions measurement technique for afterburning turbojet engines developed for AFAPL. It is used in conjunction with gas sample surveys made at the engine exhaust plane to compute the true residual emissions released to the atmosphere. PLUMOD is coded in FORTRAN IV language for the Honeywell 6000 computer.		

DD FORM 1 JAN 73 1473

EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

403468

TABLE OF CONTENTS

<u>Section</u>		<u>Page</u>
1.0	INTRODUCTION	1
2.0	COMPUTER PROGRAM DESCRIPTION	4
	2.1 The Subroutines and What They Do	6
	2.2 Variables in Common Storage	58
	2.3 Overlay Structure	102
3.0	OPERATING INSTRUCTIONS	104
	3.1 Deck Setup	104
	3.2 File Storage and Input Data Format	114
	3.3 Sample Calculation	135
	3.4 Helpful Hints	156
4.0	SOURCE LISTINGS	157
5.0	REFERENCES	457

ACCESSION for	
NTIS	Write Section <input checked="" type="checkbox"/>
DOC	B. H. Section <input type="checkbox"/>
UNANNOUNCED	<input type="checkbox"/>
J. S. I. R. I. I. I.	
DISTRIBUTION/AVAILABILITY NOTES	
CIAL	

SECTION 1.0

INTRODUCTION

Computer program "PLUMOD" described herein is an analytical model of the physical and chemical processes occurring in the mixing and reacting gas flow fields of the exhaust plumes from afterburning turbojet engines. It is an integral part of the exhaust emissions measurement technique developed for such engines for the Air Force Aero Propulsion Laboratory under Contract No. F33615-73-C-2047 (Reference 1).

Determination of the gaseous emissions from afterburning engines by sampling the exhaust gas requires overcoming problems not encountered with nonafterburning engines. These problems are a consequence of the extremely high gas temperatures at the exhaust plane, which can result in continued chemical reactions downstream of this location. If such reactions occur, exhaust plane measurements would yield emissions values which are not representative of the actual contribution to atmospheric pollution. One method of overcoming this problem is to use PLUMOD in conjunction with gas sample surveys made at the engine exhaust plane; PLUMOD predicts analytically the true residual emissions corresponding to the measurements.

This manual contains a summary of the analytical methods, a description of the computer program, and instructions for its use. Full exposition of the analytical methods used in the model, and comparisons of the model's predictions with actual survey measurements made at various axial positions in the exhaust plumes of engines, are contained in Reference 1.

The analytical model is formulated in a building-block manner, incorporating pre-existing elements where applicable. This formulation was expedient in constructing the model, but results in analytical approaches that are sometimes indirect. The most prominent pre-existing elements are: (a) the steady-state turbulent gas jet mixing analysis used in computer program JETMIX (Reference 2); (b) a formal solution of gas-phase chemical reaction kinetics equations developed by NASA-Lewis Research Center (Reference 3); (c) an approximate combustion gas kinetics analysis developed as an undocumented extension of techniques developed for hydrogen-fueled hypersonic ramjets (Reference 4). Other elements required to complete the analysis were developed specifically for this model.

Physical parameters input to the model include: ambient air properties (temperature, pressure, humidity, flight speed); fuel properties (temperature, hydrogen content, heating value); engine cycle parameters (ram air temperature, bypass ratio, main combustor fuel-air ratio and CO and NO_x emissions indices, fan discharge temperature, exhaust jet diameter); and exhaust jet survey probe data at several points (radial location, impact pressure, gas sample analysis). The gas analysis requires volumetric determination of CO and CO₂ with the sample dried to saturation at 32°F, and total HC (as single-carbon-atom molecules) and total NO_x with sample water content as-sampled.

The composition and properties of the gas flow at each probe measurement point are derived from the measurements. In this derivation, the gas composition at a point in the flow is assumed to be heterogeneous or time-varying. This assumption was chosen in preference to a homogeneous gas with nonequilibrium chemical composition, because examination of chemical reaction kinetics indicated that, under high temperature afterburner conditions, the reactions would proceed to equilibrium in a few inches. The existence of nonequilibrium concentrations of CO and HC in the gas samples is better explained by postulating that, over a part of the sampling time, the probe was immersed in a gas mixture much richer or much leaner than the mean fuel concentration of the sample.

The flow is assumed to be axisymmetric, so that the gas properties at each sample point are applied to the flow in an annular streamtube. The time-average properties of the flow in each tube are also computed.

The intermixing of the annular streamtubes and the diffusion of the jet air into the static or moving ambient, are computed by the JETMIX analysis (Reference 2), which performs a numerical solution of the differential equations of continuity of mass, momentum, energy, gas species, and turbulence kinetic energy. The flow initially in each of the streamtubes assigned to each sample point is treated as a separate gas, as is the ambient air. These gases are treated as homogeneous, inert, ideal gases with constant specific heats. Any increase in total temperature due to chemical reactions is calculated later, but it is assumed that the density change is not sufficient to influence the mixing. In an afterburner, the combustion reactions should be essentially complete upstream of the exhaust nozzle, so that the reactions continuing into the exhaust plume are only consuming trace contaminants.

A differential equation governing the formation, convection, and decay of a characteristic of the gas heterogeneity is solved numerically throughout the flow field, using the previously calculated properties of the flow. This concept was taken from the works of Spalding (References 5, 6 and 7).

The nonreacting flow field solution provided by the JETMIX analysis and the Spalding heterogeneity solution are not used directly by the plume model, but rather are used to guide the mixing between and homogenization within a fixed number of computation tubes extended from the original streamtubes assigned to each sample point. This approach is taken to minimize the number of chemical kinetics calculations. The computation proceeds in axial steps through the plume. The gas composition in each tube is modified by intertube mixing and intratube homogenization at each step. After several steps, the gas composition is further modified by chemical reactions over a time increment corresponding to the step length and local velocity.

A choice of two chemical reaction kinetics analyses is provided. One of these, GCKP, consists of a numerical solution of simultaneous differential equations formed from the forward and reverse reaction rates of a system of 23 two- and three-molecule collision reactions. Although having a high degree of rigor, GCKP is computationally cumbersome. The alternate kinetics analysis,

SCKP, utilizes approximations applicable to high-temperature combustion reactions to achieve computational efficiency. These approximations are associated with a radical pool in pseudo-equilibrium, the decay of which is paced by selected three-body reactions.

Neither chemical kinetics analysis treats consumption of unburned HC. This is done with a phenomenological technique using an empirical relation of ignition delay for kerosene as a function of temperature.

At preselected axial locations, the model performs a summation of fuel and contaminant flow in each computation tube, providing predicted values of local and overall emissions indices. When all local gas temperatures have been reduced to the point where chemical reactions cease, the emissions indices will not change with length. These are then the true residual emissions.

SECTION 2.0

COMPUTER PROGRAM DESCRIPTION

The analytical model is coded in FORTRAN IV (sometimes called FORTRAN A) language for General Electric's Honeywell 6000-series computer. The program comprises 102 subprograms, in addition to system library subroutines, and uses 166 blocks of labelled common storage. Overlay processing is used to reduce core memory requirements; the program currently requires 41K of memory.

A brief statement of the function of each subprogram follows in Section 2.1, with emphasis on the interfaces with other subprograms. The comments are intended to relate the subprogram logic to the description of analytical methods presented in Reference 1, and to aid interpretation of the program source lists in Section 4.0. The subprograms are listed in alphabetical order by deck label. Entry names may be associated with deck labels with the aid of the cross-reference chart, Table 1. The link names refer to the program overlay structure in Section 2.3 and show the general location of the subprogram in the logical structure.

Variables stored in labelled common are identified and described in detail in Section 2.2. It is presumed that the variables used internally in the various subprograms can be adequately identified through examination of the source listings in Section 4.0. The location of each common block in the overlay structure is also indicated.

Table 1. Cross-Reference List of Subprogram Entry Names (SYMDEFS).

SYNDEF	DECK	SYNDEF	DECK	SYNDEF	DECK
.....	MAINP	JETINP	JETINP	REDPRE	REDPRE
//	BLKRDA	JETPRF	JETPRF	RH2	RH2
//	MAINBD	JETPRP	JETPRP	RINP	KINPP
//	CTABPR	JMESHM	JMESHM	SCALE	SCALE
//	ELOCKT	JTCTRL	JTCTRL	SCKP	SCKP
//	NAMBLK	JTEDGE	JTEDGE	SEARCH	SERCH
//	BLKTH	JTFILE	JTFILE	SETM	MLPKG
//	KEELOK	JTINIT	JTINIT	SETUP	SETUP
//	BLKST	JTOUT1	JTOUT1	SOLV3	SOLV3
//	BLCK	JTOUTP	JTOUTS	SPALD	SPALD
ADJMC	ADJMC	JTOUTS	JTOUTS	STPROP	STPROP
AITER1	AITER	JTSTEP	JTSTP	SUMCPD	SUMCPD
AITER2	AITER	KINET	KINET	SUMUP	SUMP
ALFA2	ALFA2	KINP	KINPP	TABPRT	TABPRT
AUTO	AUTOO	LCFIT	LCFIT	TDSEO	TDSEO
CAROL	CAROL	LESV	LESVV	TFMH	TFMH
CASG	CASNM	LFIT	LFIT	TFMH1	TFMH1
CASI	CASNM	LFIT1	LFIT1	THRM	THRM
CASM	CASNM	LSPFIT	LSPFIT	THRUST	THRUST
CHEMB	CHEMB	MAIN	MAINJ	TRIDIA	TRIDIA
CINP	CUBSS	MAIN	CFILE	XSIZE	XSIZE
COC02B	COC02B	MFMAIN	MFMAIN	YOF	YOFX11
COEFF	COEFF	MOVE	MLPKG		
CUBS	CUBSS	MPRINT	PRINT		
DERIV	DERIV	MSHCUT	MSHCUT		
DERV	DERVV	MXFLT0	MXFLT1		
DFEQ	DFEQ	MXFLUT	MXFLUT		
EOGASH	EOGASH	MXKINO	MXKINO		
EOGAST	EOGAST	NEWFLC	NEWPSI		
ECKIN	ECKIN	NEWNET	NEWNET		
ERR0R1	ERR0R1	NEWPSI	NEWPSI		
ERROR1	MAINJ	NOX2B	NOX2B		
ERR0PC	ERR0PC	OUT1	OUTPP		
FILL	FILL	OUT2	OUTPP		
FMPYC	MLPKG	OUT3	OUTPP		
FRGAST	EOGAST	OUTP	OUTPP		
FUEL	FUEL	PADD	PADD		
GANCP	GANCP	PARD	PARD		
GAMH	GAMH	PBOLIC	PBOLIC		
GCKP	GCKP	PERR	PERRR		
GCKP1	GCKP1	PRAT	PRAT		
GMODFY	GMOD	PRED	PRED		
HCE02	HCE03	PRED1	PRED		
HCE02	HCE02	PROJ	PROJ		
HYCARB	HYCARB	PSEQ2	PSEQ2		
INTC	INTEE	QIREM	QIREM-		
INTE	INTEE	RATCON	RATCON		
INTG	INTG	READIN	READIN		
INTG	INTEE	READT	READT		
INTI	INTEE	READT	READT2		

2.1 The Subroutines and What They Do

Subprogram descriptions are listed below in alphabetical order by deck label. Corresponding source listings are provided in Section 4.0. Table I may be used to locate the subprogram containing a specific entry name (SYMDEF).

Subprogram ADJMC

Entry Name: ADJMC

Type: Subroutine

Link: EMIS

Function: ADJMC is called from subroutine MXKINO to initialize the raw fuel molecule weight and adjust the polynomial coefficient A_6 to an absolute enthalpy base. These procedures are described in Appendix A (Part 3) of Reference 1.

Subprogram AITER

Entry Names: AITER1, AITER2

Type: Subroutine

Link: CPLINK

Function: Not used in PLUMOD. Call is equated to ERROR1. No list.

Subprogram ALFA2

Entry Name: ALFA2

Type: Subroutine

Link: LMXF

Function: Evaluates each array element α_{ji} , which is the fraction of flow in "NEWNET" tube j at the previous axial station that is transferred by intertube mixing to tube i at the new station. The α 's are stored in common block/CMXFLT/, which also contains flows of "JETMIX" dummy gases in each tube. Fuel flows are taken from block/CNWNET/, and total flows from /CMASS/. The single argument is the number of fuel-bearing "NEWNET" tubes.

Subprogram AUTOO

Entry Name: AUTO

Type: Subroutine

Link: LGCK

Function: "Automatic elimination from error considerations of species with non-representative errors." A NASA GCKP subroutine (Reference 3) that is not used in PLUMOD. Accesses common blocks /COND/, /PQRE/, and /SKIP/.

Subprogram BLCK

Type: Block data

Link: EMIS

Function: The primary function of BLCK is to initialize the species names and molecular weights which will be used by both the GCKP and SCKP kinetics routines. Sixteen constituents are utilized in GCKP (A, NH₃, raw fuel considered inert), whereas only the first thirteen are utilized in SCKP. The constituent order is: H, O, H₂, O₂, OH, H₂O, CO, CO₂, N₂, A, NO, N, HO₂, NO₂, NH₃, C₁₀ H_{10n} (or raw fuel). The names and molecular weights are stored in labeled COMMON /SNMW/. Note that the molecular weight of the fuel is dependent on the hydrogen/carbon ratio and is initialized by subroutine ADJMC.

Subprogram BLKRDA

Entry Name: none

Type: Block Data

Link: LGCK

Function: Enters specific code number, reaction rate expression coefficients, and third body efficiencies for 23 reactions, and species excluded from integration error consideration into common blocks /LSLRV/, /REAC/, /RRAT/, and /SKIP/. In the NASA GCKP program (Reference 3), this information is read as input data.

Subprogram BLKST

Entry Name: none

Type: Block Data

Link: LGCK

Function: Enters stoichiometric coefficients for 16 species and 23 reactions in common block /SPEC/.

Subprogram BLKTH

Type: Block data

Link: PREJET, EMIS

Function: The primary function of BLKTH is to initialize the polynomial coefficients utilized by the thermodynamic routine THRM. The polynomial representation of the JANAF thermodynamic properties is discussed in Appendix A-3 of Reference 1. The data are stored in COMMON /TCOF/.

Subprogram BLOCKT

Type: Block data

Link: JETMIX

Function: JETMIX block data for turbulence parameters (Common/PROPJT/

Subprogram CAROL

Entry Name: CAROL

Type: Subroutine

Link: PREJET

Function: Computes gas sample fuel-air ratio and emissions indices from "CAROL" gas analyzer readings. The arguments, all single word variables are:

N	(independent)	H/C atom ratio of fuel
WAR	(independent)	Specific humidity of combustion air
RHC		Gas analyzer readings for HC, CO,
RCO		CO ₂ , and NO _x , mole fraction,
RCO2	(independent)	with standard sample moisture
RNOX		content.
FAR	(dependent)	Fuel-air ratio
EIHC		Emission indices for HC, CO, and NO _x ,
EICO	(dependent)	lb/klb fuel
EINOX		

The sample hydrogen content is estimated by reference to function RH2.

Subprogram CASMM

Entry Name: CASM, CASI, CASG

Type: Subroutine

Link: LGCK

Function: "Chooses the initial step (restart) formula (entry CASI) or the general step formula (entry CASG). Sets up the augmented matrix, and computes increments." This is a modified version of the NASA GCKP subroutine of the same name (Reference 3). The modifications permit a limited number of attempts to recover from negative species concentrations by the simple expedient of repeating the step with half the step size. Calls subroutine LESV, and accesses common blocks /OPTS/, /COND/, /DERN/, and /MATX/.

Subprogram CFILE

Entry Name: MAIN

Type: Subroutine

Link: CFILE

Function: Subprogram MAIN is the main subroutine of overlay CFILE. This routine is used for two purposes:

- 1) Merge the output files from PREJET (file code 1), JETMIX (file code 2), and SPALDG (file code 3) to a single output file (file code 4).
- 2) Decollate the input file (file code 7) and restore the PREJET, JETMIX, SPALDG data on file codes 1, 2, 3 respectively.

Subprogram CHEMB

Entry Name: CHEMB

Type: Subroutine

Link: EMIS

Function: Not used. Equated to system library routine ERROR. No list.

Subprogram COC02B

Entry Name: COC02B

Type: Subroutine

Link: LSCK

Function: Subroutine COC02B calculates the finite rate consumption of CO over a specified time step. The arguments to the subroutine are the static pressure and the magnitude of the time step.

The procedures for treating the CO consumption are described in Section 4.6.2 of Reference 1. Essentially, the procedure consists of a closed form integration of the rate expression for the reaction $\text{CO} + \text{OH} = \text{CO}_2 + \text{H}$. If the result indicates a CO concentration below that at mixture equilibrium, the CO mole fraction is set to the equilibrated value.

Subprogram COEFF

Entry Name: COEFF

Type: Subroutine

Link: SPALDG

Function: The function of subroutine COEFF is to evaluate the coefficients in the partial differential equation for mean square fuel concentration fluctuation intensity (G). The coefficients are calculated at the cross-stream mesh points for each axial station in the step-by-step finite difference solution procedure. Local aerodynamic properties, from subroutine STPROP, are used in the calculations. The variable K in the calling sequence for COEFF is an indicator which, in this program, is held constant (K=1).

Subprogram CTABPR

Type: Block Data

Link: Main

Function: CTABR sets the data in common /CTABPR/ to 0. (Used by subroutine TABPRT)

Subprogram CUBSS

Entry Name: CUBS, CIMP

Type: Subroutine

Link: LGCK

Function: From entry CUBS, "computes coefficients for cubic spline interpolation and differentiation from input table." From CIMP, performs one of the several interpolations and differentiations, using these coefficients, depending on value of ITPSZ in common block /AFUN/. This subroutine is taken intact from the NASA GCKP program (Reference 3).

Subprogram DERIV

Entry Name: DERIV

Type: Subroutine

Link: CPLINK

Function: Subroutine DERIV is used to calculate a partial derivative using divided differences that is compatible with the DFEQ routine. The arguments to the routine are the independent variable, the dependent variable, the calculated variable derivative, and the index limits in the independent variable table. This subroutine is called from JTSTEP to compute the partial derivative of the turbulent kinetic energy with respect to ψ .

Subprogram DERVV

Entry Name: DERV

Type: Subroutine

Link: LGCK

Function: "Computes all derivatives with respect to the independent variable" (time, in the PLUMOD application). This subroutine is taken intact from the NASA GCKP program (Reference 3). It accesses common blocks /OPTS/, /COND/, /SPEC/, /GHSC/, /DERN/, /NECC/, and /SABS/, and includes a call to PARD.

Subprogram DFEQ

Entry Name: DFEQ

Type: Subroutine

Link: CPLINK

Function: Subroutine DFEQ solves a partial differential equation of the general "diffusion" equation form using the methods described in Section 4.3.10 of Reference 1. The general "diffusion" equation form is:

$$\epsilon \frac{\partial F}{\partial X} = \alpha \frac{\partial}{\partial y} \left(\beta \frac{\partial F}{\partial y} \right) + \eta \frac{\partial F}{\partial y} + \delta F + \gamma$$

The coefficients of the differential equation are passed to the routine via COMMON/PARAM/ and COMMONPARAM1/. These commons also contain the dependent and independent variables and the boundary conditions applicable for the given partial differential equation. The first of the four arguments to the routine denotes the difference form to be used for replacement of $\frac{\partial}{\partial y} \left(\beta \frac{\partial F}{\partial y} \right)$.

The next two arguments are the lower and upper boundary condition indicators. The fourth argument is an error indicator switch. The reader is referred to the program listing for the various settings of the first three arguments.

The solution is effected by replacing the partial derivative by divided differences to obtain a set of linear simultaneous equations. The tri-diagonal system is solved using subroutine TDSEQ. The dependent variable and its derivative are returned in FM1 and DFMI located in COMMON/PARAM/.

Subprogram EQGASH

Entry Name: EQGASH

Type: Subroutine

Link: PREJET

Function: Computes thermostatic properties of combustion gas mixtures at a specified enthalpy, pressure, and fuel-air ratio. Calls EQGAST with successive estimations of temperature, using QIREM to direct convergence. Arguments are the same as EQGAST, except that T is a dependent variable and H is independent.

Subprogram EQGAST

Entry Name: EQGAST, FRGAST

Type: Subroutine

Link: PREJET

Function: Computes thermostatic properties of combustion gas mixtures at a specified temperature and pressure. From entry EQGAST, the equilibrium mixture composition at the specified fuel-air ratio is computed by a call to HCEQ2. From entry FRGAST, the mixture is specified. Arguments for entry EQGAST are:

FAR Mass ratio of fuel to dry air

WAR Specific humidity of air

HC H/C atom ratio of fuel

T Temperature ($^{\circ}$ R)

P Pressure (psia)

FIXCO } If .TRUE., CO or NO concentration is

FIXNO } predetermined and excluded from equilibrium calculations

The preceding seven arguments are independent variables.

Z Mixture composition, moles/lb mix, eleven species in order H, O, H_2 , O_2 , OH, H_2O , CO, CO_2 , N_2 , A, NO

If FIXCO = .TRUE., Z_{CO} and Z_{CO_2} must be specified.

If FIXNO = .TRUE., Z_{NO} and A_{N_2} must be specified

The following six arguments are independent variables:

H Mix enthalpy (Btu/lb)

MWT Mix molecular weight (lb/mole)

S Entropy (Btu/lb $^{\circ}$ R)

SPV Specific volume (ft^3/lb)

A Frozen sonic velocity (ft/sec)

CP Frozen specific heat (Btu/lb $^{\circ}$ R)

The arguments FAR, WAR, HC, FIXCO, and FISNG are not used for entry FRGAST. Specie thermo properties are transmitted from subroutine THRM thru block/GHSC/.

Subprogram EQKIN

Entry Name: EQKIN

Type: Subroutine

Link: LSCK

Function: Subroutine EQKIN is called from KINET to determine the equilibrium combustion efficiency (β) and temperature (T_{eq}) which will be approached by the rate expressions. The equilibrium temperature is established by iteration using subroutines HXEQ2 and QIREM. Upon convergence, the equilibrium β is computed using the known fuel air ratio and the equilibrium molecular weight generated in subroutine HCEQ2.

When P DUM (20) is COMMON/CPRING) is input as 39., the kinetics scratch file defined in COMMON/DQIREM/ is changed to 39. When this situation occurs, the equilibrium temperature iteration history is recorded on the scratch file (in addition to the initial kinetics information on file 40) (see - subroutine MXXINO).

Subprogram ERROR1

Entry Name: ERROR1

Type: Subroutine

Link: PREJET & LSCK

Function: Writes contents of block/CQIREM/, containing control info for QIREM, on file 6. Transfers up to 3000 132-character BCD records (without carriage control symbols) from file number contained in single word of /DQIRM/ to file 6. Calls ERROR to generate logic trace and terminate execution.

Subprogram ERRORC

Entry Name: ERRORC

Type: Function

Link: LGCK

Function: "Computes the relative error in an integration step of size H. Determines the controlling variable." This is a renamed version of the NASA GCKP routine ERRORR. Renaming was necessary to avoid conflict with a Honeywell system library subroutine called elsewhere in PLUMOD in the event of a computation error. Common blocks /COND/, /SINT/, /PQRE/, and /SKIP/ are accessed.

Subprogram FILL

Entry Name: FILL

Type: Subroutine

Link: INLINK

Function: This routine will fill vacancies in input lists by linearly interpolating between known values. Arrays X and Y are input along with limits NA and NB on the known values.

Vacancies in the Y list are detected by the presence of a dummy value which the list was initialized to. Any such positions are filled by interpolating between the closest known values.

Subprogram FUEL

Entry Name: FUEL

Type: Subroutine

Link: PREJET

Function: Computes estimated thermodynamic properties of raw fuel vapor. Four arguments are: Fuel H/C atom ratio, temperature ($^{\circ}\text{R}$) (both independent), enthalpy (Btu/lb), and specific heat (Btu/lb $^{\circ}\text{R}$).

Subprogram GAMCP

Entry Name: GAMCP

Type: Subroutine

Link: JETMIX

Function: GAMCP computes the ratio of specific heats (γ) and the constant pressure heat capacity (C_p) at each transverse mesh point. The subroutine arguments are the temperature, γ , C_p , R, and the mesh point indices. For $800^\circ \text{ R} \leq T \leq 3600^\circ \text{ R}$, γ is computed as a function of temperature. Below 800° R , $\gamma = 1.4$ and above 3600° R , $\gamma = 1.254$.

Subprogram GAMH

Entry Name: GAMH

Type: Function

Link: JETMIX

Function: Function GAMH is called by subroutine JETINP to compute a single value of the specific heat ratio. The relations used in GAMH are identical to those used in GAMCP.

Subprogram GCKP

Entry Name: GCKP

Type: Subroutine

Link: LGCK

Function: Main driver routine for GCKP calculations. Performs certain initialization, then calls GCKP1 to perform chemical kinetics calculations on the hot part of a gas mixture. By-passed kinetics calculations if $T \leq 1500 \text{ R}$. Tube number is only argument. Has access to common blocks /TROUBL/, /INDATA/, /JETDAT/, /GASCP/, /GASTMW/, /STCTRL/, /CRPRINT/, /COND/, and /PRIN/.

Subprogram GCKPI

Entry Name: GCKPI

Type: Subroutine

Link: LGCK

Function: This is the main program of the NASA GCKP program (Reference 3) converted to subroutine form. Changes have been made to cause printing to be suppressed except just prior to a PLUMOD print station. Additional printout can be obtained by setting PDUM(11)≠0. Chemical kinetics calculations are performed by calls to CASG, INTC, INTG, INTI, KINP, OUT1, OUT2, OUT3, PRED, PRED1, and ERRORC. A non-standard return from GCKP1 occurs if a computation malfunction is encountered. Common blocks /COND/, /SINT/, /PQRE/, /PRIN/, /STCTRL/, and /CPRINT/ are accessed.

Subprogram GMOD

Entry Name: GMODFY

Type: Subroutine

Link: GMOD

Function: Reads the PREJET data from file 1, modifies the initial profiles of Spalding heterogeneity to force rapid homogenization of tubes with a mass fraction hot gas less than a specified minimum (input thru Namelist /A/), optionally modifies the mass fraction hot gas assigned to the ambient air, then restores the altered PREJET data to file 1. No arguments. Uses common block /CBITS/.

Subprogram HCEQ2

Entry Name: HCEQ2

Type: Subroutine

Link: LSCK

Function: Computes the equilibrium composition of gaseous products of combustion of CH_n fuel and moist air as a specified temperature and pressure. The arguments are the weight ratio of fuel to dry air, the specific humidity, the hydrogen-to-carbon atom ratio of the fuel, the temperature ($^{\circ}\text{R}$), the pressure (psia), two logical variables specifying whether CO and NO are to be excluded from equilibration, and the mole fractions of species H, O, H_2 , O_2 , OH, H_2O , CO, CO_2 , N_2 , A, and NO in the mixture. If the CO concentration is to be fixed, both CO and CO_2 concentrations must be predetermined. If NO is to be fixed, both NO and N_2 concentrations must be specified. HCEQ2 uses common block /GHSC/ to transmit species thermo properties from THRMM.

Subprogram HCEQ3

Entry Name: HCEQ2

Type: Subroutine

Link: PREJET

Function: Same as HCEQ2, except that gas composition is returned as moles/pound mixture instead of mole fractions.

Subprogram HYPARB

Entry Name: HYPARB

Type: Subroutine

Link: EMIS

Function: HYPARB calculates the consumption of unburned hydrocarbon ($C_{10}H_{10n}$) contaminants in a specified reaction tube using an empirical relation of ignition delay for kerosene as a function of temperature (Section 4.6.3 of Reference 1). The hydrocarbon incipience at the upstream and current station are stored as HCINCP in COMMON/GASCMP/. When the temperature upstream is greater than $400^{\circ} R$, the ignition delay is calculated using the above described relation. A weighted hydrocarbon incipience including the raw hydrocarbons poured in from the lean streamtubes is computed at the initial station. The residence time/local ignition delay is then added to yield a new hydrocarbon incipience. When the accumulated incipience equals or exceeds 1, the fuel is burned instantaneously to CO and H_2 . The composition of the raw fuel is set to 0, and the compositions of O_2 , H_2 , CO are adjusted according to the stoichiometry of the combustion reaction. The single argument to the subroutine is the streamtube index.

Subprogram INTEE

Entry Name: INTE, INTI, INTC, INTG

Type: Subroutine

Link: LGCK

Function: This is a slightly modified version of the NASA GCKP (Reference 3) subroutine of the same name. It sets up and calls for integration of the system of differential equations, then selects a step size for the next integration. The modification is a call to ERRORC rather than ERROR, which is the name of a Honeywell library subroutine used elsewhere in PLUMOD. The integration is performed by calls to AUTO, CASG, CASI, PERR, PRED, ERRORC, and SEARCH. Common blocks/OPTS/, /COND/, /SINT/, /PQRE/, and /SPEC/ are accessed.

Subprogram INTG

Entry Name: INTG

Type: Subroutine

Link: JETMIX

Function: INTG performs integration of a given function using the trapezoidal rule. The integral is returned as the third argument of the calling sequence. The integrand and independent variable tables are supplied as the first two arguments. The final two arguments are the index limits in the independent variable table.

Subprogram JETINP

Entry Name: JETINP

Type: Subroutine

Link: INLINK

Function: JETINP is the input routine which reads card input (NAMELIST \$A) from file 5, calculates initial conditions at the jet discharge plane, and sets up initial profiles to start the numerical calculation. Initially, if TAPIN is T, the jet discharge parameters and the plume initial profiles generated in the PREJET overlay are read from file 1 (subroutine JTFILE). Following file input, JETINP reads card input from file 5 using NAMELIST \$A. After counting the number of axial print stations, the ambient parameters and jet parameters not input are calculated. Static temperature, total pressure, and jet velocity, Mach number, and turbulence energy for the jet and external environment are then available.

If profiles are not input from either PREJET or a previous JETMIX run, approximate initial station profiles of velocity, temperature, turbulence intensity, and species mole fraction are established. In the case of a restart run, the JETMIX profiles at the restart station are reclaimed by calling JTFILE. Subroutine JETPRP is then called to calculate the stream function ψ at the initial station and initialize the data in common region /PROP J/. Finally, subroutine JT or T1 is called to print the initial station data and if TAPOT is T, JTFILE is called to store the initial profiles on file 3.

Subprogram JETPRF

Entry Name: JETPRF

Type: Subroutine

Link: PREJET

Function: The function of subroutine JETPRF is to convert nozzle exit plane survey data and engine cycle conditions to input for the JETMIX and SPALDG programs. Test data is transmitted from routines READIN and MXFLTI to JETPRF, where initial conditions and physical parameters are determined for the multi-tube, non-reacting jet flow field analysis. Certain input information can be supplied to this subroutine through the NAMELIST \$A.

Subprogram JETPRP

Entry Name: JETPRP

Type: Subroutine

Link: JETMIX

Function: JETPRO is utilized to compute selected jet mixing parameters at a given print station. On the initial call, subroutines SCALE, PROPJ, and GAMCP are called to calculate the initial width of the mixing zone and values of the turbulence length scale, turbulent viscosity, specific heat ratio, and constant pressure heat capacity. The stream function coordinate ψ is then calculated by integration of equation 4.3-31 of Reference 1. For a confined mixing situation, the ψ for the streamline coincident with the outer duct wall is also determined.

After calculation of the jet properties along $\psi = 0$, the sonic line is located (if the jet is supersonic) and the entrained mass flow is determined. For a confined mixing case, the subroutine THRUST is called to determine the integrated momentum of the mixing and external streams. In this case, also, a check is made to determine if the flow is choked at the given station. If the flow is choked, and the available downstream area is less than or equal to that at the current station, the run is terminated.

Subprogram JMESHM

Entry Name: JMESHM

Type: Subroutine

Link: CPLINK

Function: Subroutine JMESHM is used to redistribute the calculation mesh in the transition/similar region of the jet. The streamlines are redistributed such that the ratio of ψ 's for any two adjacent intervals is a constant. The following relation is used:

$$\psi_1 = \frac{\Delta\psi_1 (K^{i-1} - 1)}{(K - 1)}$$

where: K = mesh constant
 $\Delta\psi_1 = (\psi_2 - \psi_1)$

and $\psi_1/\psi_e = 1$ for the last mesh point.

Subprogram JTCTRL

Entry Name: JTCTRL

Type: Subroutine

Link: CPLINK

Function: JTCTRL is the main control routine in the calculation/output overlay. Subroutine JTSTEP is called to carry out the integration of the differential equations to the current calculation station. Upon return from JTSTEP, tests are initiated to determine if the potential core or supersonic core has disappeared or a merge point has been detected (coannular/coplanar jet). If so, the current X value is inserted as a calculation station. The subroutine JETPRP is called to calculate jet properties, and if printout was requested, JTOUTP is called to print station profiles. When the potential core disappears, JMESHM is called to redistribute the calculation mesh.

Subprogram JTEDGE

Entry Name: JTEDGE

Type: Subroutine

Link: CPLINK

Function: The prime functions of JTEDGE are to locate the outer edge of the jet mixing zone and to set indicators for addition of another streamline on the next axial solution step. Normally, the outer edge of the jet is taken as the point where the difference between the local velocity and the center-line velocity is 98 percent of the maximum velocity difference across the jet. When the variable SCLD in COMMON/SCALED/ is T, the edge is taken as the point where the ambient air mole fraction is 0.98.

After determination of the edge condition, the Y coordinates corresponding to the ψ coordinates are calculated by integration of the streamfunction relation using subroutine INTG. Finally, the indicator to add a streamline is set T if the difference between the edge velocity and the velocity at the adjacent inner mesh point is greater than 0.005. The first argument to the subroutine is the current X location. The Y and ψ values of the edge are returned as the next two arguments. The streamline addition indicator is the fourth argument in the calling sequence.

Subprogram JTFILE

Entry Name: JTFILE

Type: Subroutine

Link: JETMIX

Function: JTFILE is used to store/retrieve program data on/from tape or magnetic disc files. The subroutine has four sections which serve the following functions:

1. Read input profiles and parameters defining the jet from file 1 or file 2.
2. Write input and centerline parameters on file 2. Reclaim calculated profiles from file 3 and write them on file 2.
3. Read calculated profiles for a designated station.
4. Write calculated profiles on file 2.

Subprogram JTINIT

Entry Name: JTINIT

Type: Subroutine

Link: INLINK

Function: Initialize labeled common variables to standard settings prior to input.

Subprogram JTOUT1

Entry Name: JTOUT1

Type: Subroutine

Link: INLINK

Function: JTOUT1 prints the initial conditions at the discharge plane of the jet. Included are the parameters defining the jet such as T, P, U, etc. and the initial profiles of normal coordinate, streamfunction, velocity, temperature, turbulence intensity, and species mole fractions. For the confined mixing situation, the stream function value at the outer duct surface is also printed.

Subprogram JTOUTS

Entry Name: JTOUTS, JTOUTP

Type: Subroutine

Link: CPLINK

Function: Subroutine JTOUTS (ENTRY JTOUTP) is the primary output routine for the JETMLX program. JTOUTS prints a summary print of the overall station parameters of the jet. Included in this dimensionless output are the X station location, the width of the mixing zone the ψ coordinate of the jet edge, the centerline velocity, temperature, turbulence intensity, total pressure ratio, and total temperature ratio, the location of the sonic line, and the relative entrainment ratio. For a confined mixing case, this output is followed by a summary of the dimensional parameters related to the ducted mixing problem. This output consists of the X location, the duct and centerbody coordinates, the dimensionless duct coordinate and iterated value, the pressure, flow state (subsonic, supersonic, choked), the integrated momentum (thrust) and the Mach number, velocity, and static temperature in the external unentrained inviscid stream.

Entry JTOUTP also produces a print of the dimensionless and dimensional profiles at a given calculation station. The output includes the dimensionless ψ coordinate, the stream function, the Mach number, turbulence intensity, and both the dimensionless and dimensional velocities, temperatures and total pressures. The profiles of species mole fractions are printed on a following page if a species diffusion case is being run. If an output file is requested, the profiles are temporarily saved on file 3.

Subprogram JTSTEP

Entry Name: JTSTEP

Type: Subroutine

Link: CPLINK

Function: Subroutine JTSTEP is the main calculation routine for integration of the partial differential equations described in Section 4.3.9 of Reference 1. The numerical procedures for this process are given in Section 4.3.10. The differential equations are solved sequentially in the following order:

1. X - Momentum
2. Turbulent kinetic energy
3. Species continuity (if applicable)
4. Energy

The most current values of each dependent variable are used as each equation is solved using subroutine DFEQ. At each intermediate step between calculation stations, routine JTEDGE and XSIZE are called to locate the outer edge of the jet and adjust the current X stepsize. If the potential or supersonic core has disappeared, or a merge station detected (JTEDGE), a premature return is made to subroutine JTCTRL to insert the pertinent station. Otherwise, the integration continues to the next calculation station. Additional routines called at intermediate steps are SCALE, PROPJ, and MSHCUT. The latter subroutine is called to redistribute the calculation grid only if the maximum allowable number of mesh points is exceeded.

Subprogram KEBLOK

Type: BLOCK DATA

Link: LSKK

Function: The primary function of KEBLOK is to initialize parameters utilized by the SCKP kinetics routines. Specific items preset are the rate constant Arrhenius coefficients, activation energies, and 3-rd body catalytic efficiencies for the pertinent chemical reactions described in Section 4.6.2 of Reference 1.

Subprogram KINET

Entry Name: KINET

Type: Subroutine

Link: LSCK

Function: Subroutine KINET is utilized to compute the kinetic history of the reaction streamtube over a specified time step. The calling arguments to the subroutine are the static pressure, the static enthalpy, the fuel/air ratio (lbm/lbm), and the residence time of the step. The defining equations and the calculation procedure for advancing the kinetics solution is given in Section 4.6.2 of Reference 1. The static pressure, enthalpy, and fuel air ratio are considered constant over the step.

On the initial entry, the local combustion efficiency (B) and density are established at the beginning of the time step. On the second call, subroutine EQKIN is entered to establish the equilibrium conditions which the chemical reactions will approach. The kinetics solution is then carried out in a sequence of small steps (NTSTEP in COMMON/CKINET/) using the following procedure:

1. The consumption of CO is computed by a call to COCO2B which uses the relations defined in Section 4.6.2 of Reference 1.
2. The rate expression for the 3-body reactions is integrated in closed form to determine a new combustion efficiency (B defined in terms of molecular weight).
3. The static temperature and composition of the non-equilibrium mixture is then established by iteration using subroutines PSEQ2 and QIREM.
4. Finally, the consumption of nitrogen oxide contaminants (NOX) is calculated using subroutine NOX2B.
5. The procedure is repeated until the time interval specified by the fourth argument is reached.

Subprogram KINPP

Entry Name: KINP, RINP

Type: Subroutine

Link: LGCK

Function: This is a modified version of the NASA GCKP (Reference 3) subroutine of the same name. It performs initialization and reads input data. The modifications delete most of the input, as the chemical reaction system is built into labelled common in PLUMOD, rather than being input as data, and the initial composition is determined internally from previous calculations. Namelist SPROB is read for integration controls on the first call to GCKP only. Subprograms called include CINP, CUBS, and THRM. There are no arguments. Common blocks/LTUS/, /OPTS/, /COND/, /REAC/, /RRAT/, /AFUN/, /SPEC/, /SINT/, /TCOF/, /PRIN/, /XVSA/, /SNMW/, /KOUT/, /GHSC/, /PQRE/, /SKIP/, /NECC/, /MISC/, /INDX/, /LSLRV/, /CINSAV/, and /STCTRL/ are accessed.

Subprogram LCFIT

Entry Name: LCFIT

Type: Subroutine

Link: JETMIX

Function: LCFIT is a least squares curve fitting routine utilized in the confined mixing option. After fitting a least squares parabola to the i th and $i + 1$ points (the X of interest being between X (I) and X (I+1)) this routine will return an interpolated value for the function, the derivative of the function at the point, or the integral of the function from one specified X to another.

Input to the routine consists of X and Y arrays for the points on the curve, the number of points, a list of X values at which the calculations are to be done, the constant of integration (if applicable), the number of calculation points, indicator for type of calculation, and an array of coefficients of the parabola if available from a previous call.

Subprogram LESVV

Entry Name: LESV

Type: Subroutine

Link: LGCK

Function: "This routine is a general double precision linear equation solver. In this program (GCKP) it is used to compute the increments (of the dependent variables)." It is taken intact from the NASA GCKP program (Reference 3). Common blocks /COND/ and /MATX/ are accessed.

Subprogram LFIT

Entry Name: LFIT

Type: Subroutine

Link: LMXF

Function: LFIT is used to provide linear interpolation, integration, or differentiation of a function specified in tabular form. Input to the routine consists of X and Y arrays for the points on the curve, the number of points, a list of X values at which the calculations are to be done, the constant of integration (if applicable), the number of calculation points, and an indicator for the type of calculation (0 = interpolation, -1 = integration, 1 = differentiation).

Subprogram LFITL

Entry Name: LFITL

Type: Subroutine

Link: SPALDG

Function: Same as LFIT

Subprogram LSPFIT

Entry Name: LSPFIT

Type: Subroutine

Link: LMSF

Function: Calls LFIT, and has the same arguments.

Subprogram MAINBD

Type: Block Data

Link: Main

Function: Initialize the data in the commons /TROUBL/ and /CBITS/.

Subprogram MAINJ

Entry Name: MAIN, ERROR 1

Type: Subroutine

Link: JETMIX

Function: Subprogram MAIN is the main program of overlay JETMIX. Its prime function is to load in the subordinate input and calculation overlays and call the routines JETINP, JTCTRL, and JTOUTS for input, calculation, and output respectively.

Entry point ERROR1. Sets ERR (block/TROUBL/) to T to indicate a JETMIX error and returns to the main program in link o.

Subprogram MAINP

Entry Name:

Type: Main Program

Link: Main

Function: The main program MAINP provides control for execution of the programs in the PLUMOD system. The prime functions of this routine are as follows:

1. Read the initial three case identification cards.
2. Read and print the fixed format program cards, load the proper program overlay, and call the main subroutine of the overlay.
3. Print error messages and/or the "end of job" conditions.

Depending on the "program" card TAPIN, TAPOT settings, files 1, 2, 3, 4, or 7 are rewound prior to entry of the main subroutine of the called overlay. Upon return, end of file marks are written on the generated output files.

When an error condition (ERR = T) is returned by JETMIX or SPALDG, the program CFILE is called to merge the partial output files such that the job can be restarted.

Subprogram MFMAIN

Entry Name: MFMAIN

Type: Subroutine

Link: LMKF

Function: MFMAIN provides control for the calculation of the mixing/homogenization of the heterogeneous gas streams. On the initial entry, NAMELIST \$INPUT is read and mixing step parameters are initialized. Subroutine READT is then called to read JETMIX and SPALDG data from files 2 and 3 and initialize aero-data regions. A maximum of five data stations may be retained in memory at one time.

The solution is advanced stepwise in the axial direction by consecutively calling NEWNET to generate reaction tubes and MXFLUT to perform the mixing/homogenization process. The number of mixing steps before return for a kinetics calculation is specified by NIST in COMMON/STCTRL/. Return also occurs when an overall print station is reached. In this instance, FINAL in COMMON/STCTRL/ is set T to direct control to the output overlay. If the current X location goes beyond the midpoint of the data stations in memory, the JETMIX and G profiles are shifted upstream and READT is recalled to read new station information. Stepwise printout of the mixing steps occurs if PDUM (18) in COMMON/CPRINT/ is non-zero.

Subprogram MLPKG

Entry Names: FMPYC, MOVE, SETM

Type: Subroutine

Link: MAIN

Function: A GMAP subroutine drawn from the AEG Scientific Library to perform high-speed manipulation of arrays in core memory. Cannot be coded in FORTRAN because of variable number of arguments. No list included.

FMPYC is used for floating-point multiplication of a constant times the elements of an array and storage of the resultant products.

The first argument is the number of arrays to be multiplied by the constant. The second is the constant to be used. Then follows for each array, the array to be multiplied, the array in which to store the products, and the number of elements in the array to be multiplied.

MOVE moves data from one array to another or shifts data within the confines of the same array.

The first argument in the calling sequence is the number of arrays to be moved. There are then four arguments for each array to be moved. The first of these is the initial location of the array to be moved. The second is the destination of the array to be moved. The next is the number of elements of the array to be moved. The fourth is the storage increment between successive elements of the array.

If data are to be shifted upward within an array, the number of elements should be negative.

SETM sets regions of memory to a specified value.

The first argument is the number of arrays to be set to a given value. The second is the data word to be used. Then follows for each array the beginning of the array and the number of elements of the array to be set to the specified value.

Subprogram MSHCUT

Entry Name: MSHCUT

Type: Subroutine

Link: CPLINK

Function: MSHCUT is used to redistribute the calculation mesh if the number of points exceeds the maximum (NMAX - COMMON/INPJET/). The arguments to the routine are the region indicator, the stream-function, and the number of profile points. The number of mesh points is reduced by NRED (COMMON/UMESH/).

For the potential core region, a new semi-uniform mesh is calculated. In the transition/similar region, a new array of stream function coordinates is produced such that the ratio of $\Delta\psi$'s for any two adjacent intervals is a constant.

Subprogram MXFLTI

Entry Name: MXFLTO

Type: Subroutine

Link: PREJET

Function: Establishes the detailed properties of the two-part model of the heterogeneous gas flow from the gas analysis and impact pressure measurements made with a probe at one point in the engine exhaust plane. The probe point number is the single variable in the calling sequence; the remaining input data are taken from common block /INDATA/. Calculated properties are stored in blocks/JETDAT/ and /GASCMP/. Scratch file 40 is used to store iteration histories for subsequent dumping by subroutine ERROR1 in the event of convergence failure. Utilizes subroutines RH2, FUEL, PRAT, CAROL, EQGASH, EQGAST, FRGAST, and several utility subroutines.

Subprogram MXFLUT

Entry Name: MXFLUT

Type: Subroutine

Link: LMXF

Function: Performs the intertube mixing and intratube homogenization over an axial step to define flow, fuel concentration, enthalpy, and mixture composition of the hot and cold parts of each "NEUNET" tube. Uses ALFAZ to define the intertube mixing coefficients. Calls HYCARB to compute HC consumption in the step. Has no arguments, but uses common blocks /CREACT/, /CMXFLT/, /CSPECI/, /CNWNET/, /CCHECK/, /JETDAT/, /GASCMP/, /GASTMW/, /CMASS/, /CPRINT/, /STCTRL/, /CAXIAL/, and /CAGAIN/.

Subprogram MXKINO

Entry Name: MXKINO

Type: Main Program

Link: EMIS

Function: MXKINO provides control for computation of the mixing/homogenization of the heterogeneous gas streams, the subsequent chemical kinetic calculations in the hot (fuel rich) portion of each reaction tube, and finally the computation and output of the plume emission indices at specified streamwise measuring stations.

Initially, the fuel molecular weight and adjusted thermo coefficient for the raw fuel are established by a call to ADJMC, data regions are initialized, and control input NAMELIST/A/ is read.

The computation for each streamtube then proceeds downstream by successively performing mixing/homogenization steps (link LMXE) and chemical kinetic steps (links LGCK, LSCK). After each mixing step, the molecular weights are computed and the static temperatures are established by a call to subroutine TFMH. The chemical kinetics step is accomplished either by use of the NASA-GCKP program (Section 4.6.1 of Reference 1) or the GE-SCKP program (Section 4.6.2). Prior to initiation of a chemical kinetics step, scratch file 40 is rewound and initial data are saved in the event of a calculation malfunction. When the calculation has reached a streamwise measuring station, as given by the PRINT variable in COMMON/OPCTRL/, link LFIN is entered to provide computation and display of the residual emission indices. The calculation is terminated when the final measuring station is reached.

Program errors are detected if the variable ERR in COMMON/TROUBL/ is set T. When this situation occurs, the last generated kinetic history is printed by a call to subroutine ERROR1.

Subprogram NAMBLK

Entry Name: None

Type: Block Data

Link: LGCK

Function: Enters alphanumeric data in common blocks /LTUS/, /OPTS/, /SPEC/, and /KOUT/. In the NASA GCKP program (Reference 3), these data are used as input.

Subprogram NOX2B

Entry Name: NOX2B

Type: Subroutine

Link: LSCK

Function: Subroutine NOX2B calculates the rate of formation of NO by integration of the rate expressions described in Section 4.6.2 of Reference 1. The single rate equation, developed by assuming pseudo equilibrium concentrations of N₂, O₂, OH, O, H, and "steady state" concentration of N is integrated numerically using a Runge-Kutta technique.

Subprogram NWNET

Entry Name: NWNET

Type: Subroutine

Link: LMXF

Function: Defines boundaries of "NWNET" tubes, which are tubes of constant fuel flow, by interpolation in the "JETMIX" flow field solution arrays. Computes flows of "JETMIX" dummy gases in each tube, and mean velocity, heterogeneity and reaction time. Has no arguments, but uses common blocks /CAGAIN/, /CINPJT/, /JETDAT/, /CCHECK/, /CNWNET/, /CSPECI/, /CINPUT/, /CAXIAL/, /CLOCAL/, /CMXFLT/, /CMASS/, /GASTMW/, and /CRRINT/.

Subprogram NWPSI

Entry Name: NWPSI, NEWFLO

Type: Subroutine

Link: LMXF

Function: Establish a set of values of normalized stream function, ranging from 0.0 to 1.0, so that each "NWNET" tube will contain approximately the same number of points. The 50-element array containing the stream functions is the first argument of the NWPSI calling sequence; the second is the number of stream functions in the set. The flows in the "NWNET" tubes are taken from common block /CMASS/. At the initial computation station, these flows are established through entry NEWFLO, which has no arguments, using radii, velocities, and specific volumes from block /JETDAT/.

Subprogram OUTPP

Entry Name: OUTP, OUT1, OUT2, OUT3

Type: Subroutine

Link: LGCK

Function: This is a modified version of the NASA GCKP (Reference 3) subroutine of the same name. The modifications were made for a GE-AEG stand-alone version of GCKP, and have no effect in the PLUMOD application. OUTPP performs unit conversions and prints the various pages of output. Output data are obtained from common blocks /OPTS/, /COND/, /SINT/, /KOUT/, /REAC/, /RRAT/, /AFUN/, /SPEC/, /XVSA/, /NECC/, /PQRE/, /DERN/, /SKIP/, /GHSC/, and /SABS/.

Subprogram PADD

Entry Name: PADD

Type: Subroutine

Link: CPLINK

Function: PADD is used to add streamlines to the calculation mesh at the edge of the jet. This routine is called from JTSTEP if the JTEDGE subroutine detected that points were to be added at the next station due to entrainment of the ambient stream.

In the potential core region, one point is added at the same spacing as the adjacent two outer points. In the transition/similar region, a new point is added based on the constant ratio rule for $\Delta\psi$'s of adjacent intervals.

Subprogram PARDD

Entry Name: PARD

Type: Subroutine

Link: LGCK

Function: "Computes all mixed partial derivatives." This subroutine is taken intact from the NASA GCKP program (Reference 3). It accesses common blocks /OPTS/, /COND/, /SPEC/, /REAC/, /RRAT/, /GHSC/, /NECC/, /SABS/, and /DERN/.

Subprogram PBOLIC

Entry Name: PBOLIC

Type: Subroutine

Link: SPALDG

Function: The function of subroutine PBOLIC is to set up the system of simultaneous finite difference equation (for the cross-stream mesh points at an axial station) in the step-by-step solution of the parabolic partial differential equation for mean square fuel concentration fluctuation intensity (G). A generalized Crank-Nicolson finite difference representation is used, allowing either explicit or implicit formulation of the equations. However, the solution procedure, in subroutine TRIDIA, assumes an implicit formulation (simultaneous equation) in either case, so that there is no advantage to be gained by utilizing the capability for explicit formulation. A tridiagonal system of simultaneous linear algebraic equations is established, utilizing coefficients calculated in subroutine COEFF. The resulting set of equations is then solved in subroutine TRIDIA. The variable K in the calling sequence for PBOLIC is an indicator which, in this program, is held constant ($K = 1$).

Subprogram PERRR

Entry Name: PERR

Type: Function

Link: LGCK

Function: This is a slightly modified version of the NASA GCKP (Reference 3) subprogram of the same name. It predicts the computation error that can be expected from using given numerical integration step size. The modification is a call to ERRORC rather ERROR, which is the name of a Honeywell library routine used elsewhere in PLUMOD. Common blocks /COND/, /SINT/, /DERN/, and /PQRE/ are accessed.

Subprogram PRAT

Entry Name: PRAT

Type: Function

Link: PREJET, LFIN

Function: Calculates ratio of probe impact pressure to local static pressure for a perfect gas of specified Mach number (subsonic or supersonic) and specific heat ratio, which are the two arguments.

Subprogram PREDD

Entry Name: PRED, PRED1

Type: Subroutine

Link: LGCK

Function: This subprogram is taken directly from the NASA GCKP program (Reference 3). It "performs all necessary pre-derivative calculations." Common blocks /OPTS/, /COND/, /SPEC/, /REAC/, /RRAT/, /GHSC/, /NECC;, and /SABS/ are accessed.

Subprogram PRINT

Entry Name: MPRINT

Type: Subroutine

Link: LMXF

Function: PRINT is the output routine for the mixing/homogenization program, called from MFMAIN when PDUM(1B) [COMMON/CRRINT/] is non-zero. Items printed are the current axial location, and the radius, fuel flow, total mass flow, velocity, mass fraction of rich flow, fuel/gas ratios (rich and lean), Spalding G function, and residence time for each reaction streamtube.

Subprogram: PROPJ

Entry Name: PROPJ

Type: Subroutine

Link: JETMIX

Function: PROPJ computes the properties of a laminar or turbulent jet. The parameters determined are the density, dynamic viscosity, turbulence length scale, eddy viscosity, diffusion parameter and k_{eff}/C_p . The latter four properties are used for calculation of the turbulent mixing flow field. No computed quantities are returned through the call sequence.

Subprogram: PSEQ2

Entry Name: PSEQ2

Type: Subroutine

Link: LSCK

Function: Computes the psuedo-equilibrium composition of gaseous products of combustion of CH_n fuel and moist air at a specified temperature and degree of dissociation. The arguments are the weight ratio of fuel to dry air, the specific humidity, the hydrogen-to-carbon atom ratio of the fuel, the temperature ($^{\circ} R$), the double precision reaction efficiency β (Reference 1, Appendix A-2), two logical variables specifying whether CO and NO are to be excluded from pseudo-equilibrium, and the mole fractions of species, H, O, H_2 , OH, H_2O , CO, CO_2 , N_2 , A, and NO in the mixture. If the CO concentration is to be fixed, both CO and CO_2 concentrations must be predetermined. If NO is to be fixed, both NO and N_2 concentrations must be specified. PSEQ2 uses common block /GHSC/ to transmit species thermo properties from THRM.

Subprogram: QIREM

Entry Name: QIREM

Type: Subroutine

Link: PREJET, LSCK

Function: QIREM is an iteration routine for determining the roots of functions with maxima or minima using a quadratic interpolation for the root evaluation.

With each call to this routine, an independent coordinate X and its corresponding functional value Y are supplied. Based on the last three such points, a new trial X at which the desired Y is estimated to occur is calculated.

Input is supplied to the routine through a calling sequence

CALL QIREM (X, Y, YJP, QV)

and a labeled common

COMMON /CQIREM/ YTOL, YO, DYDX, CTRMAX

The input variables are:

X = trial value of x.

Y = y calculated for the current trial value of x.

XJP = x-jump to be taken before the desired y has been spanned if DYDX = 0. The sign is for a positive error; that is, the x-jump is XJP if $(Y-YO) > 0$ and the x-jump is -XJP if $(Y-YO) < 0$. XJP also indicates the branch of the curve for which the solution is to be found, if two solutions exist. XJP will be positive for the branch with negative slope, as in the sketch on page 1; XJP will be negative for a branch with positive slope.

QV(1) = counter which must be set to zero before the first entry.

YTOL = convergence tolerance on $(Y-YO)$.

YO = given ordinate, y_0 , for which the solution $y=f(x)$ is desired.

DYDX = optional estimate of the curve slope. A non-zero value should be supplied if a maximum point of a solution near a maximum point is being sought.

CTRMAY = maximum number of trials before yielding to defeat. Twenty-five trials will be allowed if CTRMAX=0.

The output from the routine is:

X = next trial value of x.

QV(1) = 0. if the solution has been found, $|Y-Y_0| < Y_{TOL}$;

QV(5) = 0. if no solution can be found but the maximum point has been found within Y_{TOL}; otherwise QV(5) is non zero.

QV, known as the quire vector, must be dimensioned by at least 8. It contains the counter, three previously tried values of x, a second counter, and the three corresponding values of y.

The input supplied via the labeled common is, to a certain extent, optional as indicated below:

1. The value of CTRMAX needs to be specified only if the limiting number of trials is not 25.
2. If DYDX and Y₀ are not specified (i.e. DYDX=Y₀=0.), the QIREM routine operates exactly the same as subroutine QIRE (although, the calling sequences to QIREM and QIRE are different).
3. Y_{TOL} must be specified unless the test for convergence is performed before calling QIREM.

The parameter DYDX indicated which of two options are desired. If DYDX=0 the iteration proceeds without reference to the local calculated values of slope until a solution value or maximum point has been passed. Then the quadratic (or linear in the case of two available points) interpolation is performed to obtain subsequent x trial values. This particular option has utility when the curve has irregularities of the type illustrated below; in this case the direction of the x-jump is determined by the ordinate error. (See the definition of XJP on the previous page).

Subprogram: RATCON

Entry Name: RATCON

Type: Subroutine

Link: LSCK

Function: Subroutine RATCON is utilized to compute the rate constants for the 3-body recombination reaction scheme described in Section 4.6.2 of Reference 1. The pressure is input to the routine via the calling sequence. The temperature is obtained as the third variable in COMMON/PSEQ/.

In the process of evaluating the rate constants, the effects of the hydroperoxyl radical are included using a "steady state" assumption for the HO₂. The rate constant for the reaction $H+O_2+M=HO_2+M$ is then adjusted using a sequence of four 2-body shuffle reactions involving the HO₂ radical. All forward rate constants are calculated as a function of temperature using Arrhenius type relations of the form:

$$k_f = AT^n e^{-(\Delta E_A/RT)}$$

Reverse rate constants for the pertinent 2-body reactions are expressed in terms of the forward rate constants using the equilibrium constants calculated from the Gibbs' Free Energy (Subroutine THRMM).

Subprogram: READIN

Entry Name: READIN

Type: Subroutine

Link: PREJET

Function: Initializes, then reads engine test data and cycle parameters from file 05 via namelist TSTDAT. Sets data to standard values if not input. Sorts and stores data in common blocks /INDATA/, /JETDAT/, /GASCMP/, and /OPCTRL/. Calls MXFLTO to initialize heterogeneous flow model at each sample location. Calculates ambient air properties using EQGAST.

Subprogram: READT

Entry Name: READT

Type: Subroutine

Link: SPALDG

Function: The function of subroutine READT is to read PREJET and JETMIX output files to obtain aerodynamic properties used in the calculation of mean square fuel concentration fluctuation intensity (G). The radial profile of G at the nozzle exit plane and mean fuel concentrations in the constituent gases are obtained from the PREJET output file. Physical constants describing the jet, as well as radial profiles of aerodynamic properties at several axial stations in the jet, are obtained from the JETMIX output files. Radial profiles of mean fuel concentration are calculated (from constituent gas concentration profiles) in READT and stored along with the profiles of axial velocity, turbulent kinetic energy, density and turbulence length scale. The profiles are retained in "current" storage, in groups of five stations at a time. As the calculation of G moves downstream, profiles at upstream stations are removed from current storage and replaced with profiles from stations further downstream. The variable JENTRY in the calling sequence indicates whether or not READT has been called previously:

IENTRY = 0 first call to READT

IENTRY > 0 READT has been called previously

Subprogram: READT2

Entry Name: READT

Type: Subroutine

Link: LMXF

Function: READT is called from MFMAIN to read JETMIX and SPALDG station profiles for use in the mixing calculation. On the initial call, subroutine REDPRE and NEWFLO are called to reestablish the nozzle discharge conditions and preset the flow rates in the reaction tubes. Initially, five sets of profiles are read, including ψ , Y, U, T, G, Mach number, and species mole fractions. The outer edge of the jet is established and a new set of profiles are interpolated using a uniform grid with a maximum of 50 points. After the radii and ψ are redimensionalized (Section 4.5.1 of Reference 1), the profiles are moved to the MXFLUT common region /CINPUT/. On successive entries, only a single set of station profiles are read (see - MFMAIN).

Subprogram: REDPRE

Entry Name: REDPRE

Type: Subroutine

Link: LMXF

Function: Subroutine REDPRE is used to restore the commons defining the jet discharge plane conditions and the jet properties generated in the PREJET overlay. The data are stored in COMMON/INDATA/, /JETDAT/, /GASCMP/, and /OPCTRL/. Before return, the temperature of the rich gas fraction of each streamtube is calculated by a call to TFMH1 and stored in COMMON/GASTMW/.

Subprogram: RH2

Entry Name: RH2

Type: Function

Link: PREJET

Function: Estimates mole fraction H₂ based on the measured mole fraction CO in the gas sample, which is the single argument of the function.

Subprogram: SCALE

Entry Name: SCALE

Type: Subroutine

Link: JETMIX

Function: The prime function of subroutine SCALE is to determine the boundaries of the mixing zone for calculation of the turbulence scale in subroutine PROPJ. For the single free jet, the width of the mixing zone is normally determined using the jet edge (subroutine JTEDGE) and the point where the axial velocity deviates from a constant value (potential core). When the variable SCLD in common /SCALED/ is T, the inner edge is determined as the point where the n-th constituent concentration has reached the value ALXLIM by turbulent diffusion. This latter parameter is also stored in COMMON /SCALED/. For a coannular/coplanar jet, the variable MERGE is checked to determine whether the solution is upstream or downstream of the merge station. If upstream, the coordinates at the edges of both the primary and secondary jets are set as well as the widths of the mixing zones to the two jets. If

Subprogram: SCALE (Continued)

downstream of the merge point, the boundaries of the mixing zones are determined using linear curve fits for the nozzle-merge station lines.

The arguments to the subroutine are the flow-field variable (U, α) to be used for determination denoting a single or coannular/coplanar jet, the axial region (core, transition, similar), and the current X station. No computed quantities are returned through the call sequence.

Subprogram: SCKP

Entry Name: SCKP

Type: Main Program

Link: LSCK

Function: Subroutine SCKP is the main control program for the high temperature Special Chemical Kinetics Procedure for calculation of the chemical reactions in a given streamtube. On entry, the initial temperature of the rich fraction of the streamtube is obtained from COMMON/GASTMW/. If less than 1500°R , the reactions are assumed quenched and control is returned to the calling program. If greater than 1500°R , subroutine THRM is called to compute the dimensionless enthalpy of the individual species. The constituents N , HO_2 , NO_2 , NH_3 , and C_nH_{2n} are not evaluated directly in the kinetics procedure. To assure conservation of energy, these species are removed by splitting the mixture into an active fraction and an inert fraction. After adjusting the composition and the static enthalpy of the active fraction, subroutine KINET is called to compute the kinetics step over the time interval specified by variable TAU in COMMON/GASTMW/. Upon completion of the step, the static temperature and composition are moved into the upstream storage locations and return is made to the calling program. The single argument to the subroutine denotes the streamtube index.

Subprogram: SERCH

Entry Name: SEARCH

Type: Subroutine

Link: LGCK

Function: "This routine uses an optimal sequential search technique to find X in the interval (A,B) such that $F(X) = FOFX$." H is taken intact from the NASA GCKP program (Reference 3). F, FOFX, A, X, and B are arguments. Common block /SINT/ is accessed.

Subpogram: SETUP

Entry Name: SETUP

Type: Subroutine

Link: SPALDG

Function: The function of subroutine SETUP is to set up JETMIX output profiles of aerodynamic properties for interpolation along streamlines in the calculation of mean square fuel concentration fluctuation intensity (G). A cross-stream linear interpolation is performed, at each axial station for which JETMIX output is available, to establish the local aerodynamic properties on the streamlines used in the finite difference solution for G. The property matrices are then transposed to facilitate interpolation along the individual streamlines, so that local properties can be determined at intermediate axial stations. The variable IRST in the calling sequence indicates whether or not the JETMIX output profiles in current storage include the initial profile (X=0):

IRST = 0 initial station is in current storage

IRST > 0 initial station is not in current storage

The variable K is an indicator which, in this program, is held constant (K = 1).

Subprogram: SOLV3

Entry Name: SOLV3

Type: Subroutine

Link: PREJET, LSCK

Function: Solves a set of three simultaneous linear algebraic equations:

$$\sum_{j=1}^3 A_{ij} X_j = B_i \quad i = 1, 2, 3$$

The vectors A, B, and X are the first three arguments of the calling sequence (X is the solution vector). The fourth argument is a logical variable which is returned .TRUE. if no solution exists.

Subprogram: SPALD

Entry Name: SPALD

Type: Subroutine

Link: SPALDG

Function: SPALD is the main program in link SPALDG, for the calculation of mean square fuel concentration fluctuation intensity (G) in the jet. SPALD controls the step-by-step finite difference procedure used to solve the parabolic partial differential equation for G, and performs all output functions. Certain input information (constants) can be supplied to this program through the namelist /CHANGE/.

Subprogram: STPROP

Entry Name: STPROP

Type: Subroutine

Link: SPALDG

Function: The function of subroutine STPROP is to calculate local aerodynamic properties, for the cross-stream mesh points, in the step-by-step solution procedure used in determining the distribution on mean square fuel concentration fluctuation intensity (G) in the jet. The properties are determined by linear interpolation along streamlines between the axial station for which JETMIX output profiles are available. STPROP uses the transposed property matrices, created by subroutine SETUP, for interpolation. The variable K in the calling sequence is an indicator which, in this program, is held constant ($K = 1$).

Subprogram: SUMCPD

Entry Name: SUMCPD

Type: Function

Link: CPLINK

Function: Subroutine SUMCPD is called from JTSTEP to calculate the diffusion flux term used in the solution of the energy equation. The single argument of the function is the local mesh point index.

Subprogram: SUMP

Entry Name: SUMUP

Type: Subroutine

Link: LFIN

Function: At specified axial stations, computes the mean properties of the flow in each NEWNET tube, and synthesizes the predicted values of gas sample analysis. Integrates the fuel and contaminant flows over all tubes and computes overall emissions indices. Prints a table of calculated data, and punches profile information on cards suitable for subsequent plotting or other postprocessing by the user. Uses routines TFMH1 and PRAT, and derives data from common blocks /GASCMP/, /OPCTRL/, /JETDAT/, /COAXIAL/, /CMASS/, /INDATA/, /STCTRL/, and /CBITS/.

Subprogram: TABPRT

Entry Name: TABPRT

Type: Subroutine

Link: MAIN

Function: TABPRT is used for labeled diagnostic printout of tabular data. The first argument in the calling sequence is the name of the array to be printed. The second is the location of the array. The third is the number of elements and the last is the number of columns to be used for the print format. Through the common region CTABPR is also input the location of the first element of the array to be printed (1st, 2nd, ect.).

The structure of the elements of the array are checked against various criteria to determine whether they are floating point numbers, integers, BCD, or octal. The formats for the printout are then set accordingly.

Subprogram TDSEQ

Entry Name: TDSEQ

Type: Subroutine

Link: CPLINK

Function: TDSEQ is used to solve a system of tri-diagonal linear equations. The coefficients of the system are passed to the routine via the first argument of the CALL sequence. The solution is returned as the first column of the array. The last three arguments are the number of equations, the maximum column size, and an error indicator.

Subprogram TFMH

Entry Name: TFMH

Type: Subroutine

Link: EMIS

Function: Subroutine TFMH is utilized to compute the static temperature and constant pressure heat capacity of the rich and lean fractions of a given stream tube. The subroutine is logically identical to TFMH1. However, species compositions and enthalpies are picked up from COMMON/GASCMP/ and calculated temperatures and heat capacities are stored in COMMON/GASTMW/. The single argument specifies the upper value of the reaction tube index. (See description of COMMON/GASCMP/.

Subprogram TFMH1

Entry Name: TFMH1

Type: Subroutine

Link: EMIS

Function: TFMH1 is used to determine the static temperature and the constant pressure heat capacity given a mixture composition and static enthalpy. The temperature is determined by iteration using subroutine THRM to calculate a trial enthalpy for comparison with the input enthalpy. If the iteration does not converge to a tolerance of 0.01°R within 30 trials, an error message is printed and current values are returned. The arguments to TFMH1 are the concentration (moles i/lbm mixture), the static enthalpy (BTU/lbm), the calculated temperature and heat capacity. Single values of the latter two parameters are returned via the call sequence.

Subprogram THRMM

Entry Name: THRM

Type: Subroutine

Link: PREJET, EMIS

Function: Computes dimensionless thermodynamic properties of up to 25 chemical species at the specified temperature, using polynomial coefficients from common block /TCOF/. The temperature (Kelvins) is the first argument of the calling sequence. If the second argument has value 0.0, only enthalpy is computed. The number of species is taken from block /COND/, which also contains logical variable NEXT which is set .TRUE. if the temperature is too far out of the range of the polynomial coefficients. If the temperature is within limits, the dimensionless enthalpy, free energy, entropy, specific heat, and derivative of specific heat with respect to temperature are computed and stored in block /CHSC/.

Subprogram THRUST

Entry Name: THRUST

Type: Subroutine

Link: JETMIX

Function: For the confined mixing case, subroutine THRUST computes the integrated momentum (thrust) at a given station. If the mixing zone has intersected a plug or center-body, the program is terminated.

Subprogram TRIDIA

Entry Name: TRIDIA

Type: Subroutine

Link: SPALDG

Function: The function of subroutine TRIDIA is to solve the tridiagonal system of simultaneous linear algebraic equations formed by implicit finite difference formulation of the partial differential equation for mean square fuel concentration fluctuation intensity (G). A set of simultaneous equations is formed for the cross-stream mesh points at each axial station in the step-by-step solution procedure. The variable J in the calling sequence is an indicator which, in this program, is held constant ($J = 2$), while the variable NN is equal to the current number of streamlines minus one.

Subprogram XSIZE

Entry Name: XSIZE

Type: Subroutine

Link: CPLINK

Function: XSIZE controls the integration step size in the streamwise (axial) direction. In the potential core region, the step-size is proportional to the width of the mixing region with the magnitude set by the value of the input variable CXPC. In the transition and similar regions, DX is proportional to the radius of the jet, with a magnitude dependent on input CXTP.

The arguments to the routine are the calculated step size, the current X location, the reference scale, the pertinent region, and a terminal step indicator (LAST). LAST is set T if the step size would go beyond a calculation station. In this case, the returned step size is reduced to identically hit the calculation station.

Subprogram YOFXII

Entry Name: YOF

Type: Function

Link: JETMIX, LSCK

Function: YOFX performs a linear interpolation in a specified table of X, Y values. The additional call sequence parameters are the desired value of X^1 and the index limits in the X table.

The I index in the common YOFXI is an estimate of the interval that X lies in (between X (I-1) and X(I)) and the IA and IB indexes are limits on the interval from which points can be used for the interpolation.

The routine will extrapolate linearly if the specified X is outside the range indicated.

2.2 Variables in Common Storage

Descriptions of the contents of blocks of labelled common follow, in alphabetical order, by block name. Within each block, variables are listed according to position occupied in the block. In some cases, the variable name may differ between routines and a typical name is given. Pertinent dimension and type information are included with the variable name (R = Real, I = Integer, L = Logical). Variables normally containing BCD data are typed as H = Hollerith, even though they may have real or integer names.

Block Name (Link)	Words	Typical Variable Name	Type	Dim.	Description
ACONVG (JETMIX)	1-100	YCD	R	100	Confined mixing data storage Jet edge at each calculation station, in.
	101-200	PD	R	100	Static pressure at each calculation station, psia.
	201-300	INDC	H	100	Flow indicator SUBSON SUPSON CHOKED
	301	CHOKE	L	1	Choke indicator T - Flow choked
	302	CHOKED	L	1	Auxiliary choke indicator T - denotes previous choked station.
ADAM01 (MAIN)					Identification block
	1-10	NAME	H	10	User name
	11-20	ADDRES	H	10	User address
	21-30	TITLE	H	10	Case title
	31-40	IDENT	M	10	Not used
ADAM02 (MAIN)	1	ENDJOB	L	1	T = case has terminated normally
	2-4	DUMMY	R	3	Not used
BC (JETMIX)					Current boundary conditions at edge of jet
	1	UEDGE	R	1	u/u_j at edge of jet (velocity)
	2	EEDGE	R	1	e/e_j at edge of jet (turbulent kinetic energy)
	3	THEDGE	R	1	T/T_j at edge of jet (temperature)
BCO (JETMIX)					Coannular jet - Primary jet boundary conditions
	1	UO	R	1	u/u_j in core of secondary jet (velocity)
	2	EO	R	1	e/e_j in core of secondary jet (turbulent kinetic energy)
	3	THO	R	1	T/T_j in core of secondary jet (temperature)

Block Name (Link)	Words	Typical Variable Name	Type	Dim.	Description
BCMIX2 (JETMIX)					Confined mixing - communication
	1	GRADU			
	2	TW			
	3	MUW			
	4	RHOW			Not Used
	5	PTE	R	1	Inviscid region stagnation pressure, psia
	6	TTE	R	1	Inviscid region stagnation temperature, °R
BCMOL (JETMIX)					Boundary conditions for species diffusion equations
	1-12	ALEDGE	R	12	Mole fractions at edge of jet
	13-24	ALO	R	12	Mole fractions in core of secondary jet
CADCNT (SPALDG)					
	1	MAX	I	1	Maximum number of equally spaced streamlines (mesh points) for finite difference calculation
	2	NS1	I	1	Number of JETMIX streamlines at initial axial station (x=0)
	3	NXS	I	1	Number of JETMIX output stations currently in use for determination of aerodynamic property profiles (interpolation)
	4	NSZMAX	I	1	Maximum number of equally spaced streamlines required for JETMIX output stations in current usage
	5	NSZ1	I	1	Number of equally spaced streamlines at initial axial station (x=0)
CAGAIN (EMIS)					Used by NEWNET
	1	AGAIN	I	1	Set = -1 after first call to avoid reinitialization
CARRY (JETMIX)					JETMIX new case indication
	1	NEW	L	1	T = new case - causes initial values to be set by JTINIT

Block Name (Link)	Words	Typical Variable Name	Type	Dim.	Description
CAXIAL (EMIS)					Axial distances in jet diam- eters (from block CINPJT)
	1	X	R	1	Present computation station
	2	XL	R	1	Next pring station
	3	ALOGX	R	1	$\ln(x+1)$
CBITS (MAIN)					Utility common
	1	BITS	R	1	GE junk work 03777777777
	2	BLANK	H	1	Hollerith blank 6hbbbbbb
CBNDRY (SPALDG)					
	1	CX	R	1	Current axial step-size ($\Delta x/d_p$) for finite difference cal- culation of "G"
	2	DXPRN	R	1	Axial distance ($\Delta x/d_p$) between stations at which "G" profiles are printed
	3	JT	I	1	Indicator (in this program, (JT \equiv 1)
CBODY (JETMIX)					Confined mixing - communication
	1-100	YCB	R	100	Coordinates of centerbody, in.
	1010200	CLSPCB	R	100	Coefficients for curve fit of centerbody
	201	YCB1	R	1	Local normal coordinate on centerbody
	202	UCL1	R	1	Local velcoity (u/u_j) on centerbody
CCHECK (EMIS)					Continuity check parameters used in NEWNET and MXFLUT
	1-12	SPECIF	R	12	Total flow of each dummy gas at current station (diagnostic only)
	13-24	WKTOT	R	12	Total flow of each JETMIX dummy gas in initial jet.
	25-36	RATIO	R	12	Not used

Block Name (Link)	Words	Typical Variable Name	Type	Dim.	Description
CCOCO2 (LSCK)					Communication block between KINET and COCO2B
	1	CORATE	L	1	T = co-kinetics f = co-pseudo-equilibrium
	2	XCOI	R	1	Initial CO mole fraction
CCONST (SPALDG)					
	1	CONST1	R	1	Constant (= $\alpha_t \sqrt{g_c J_e}$)
	2	CONST2	R	1	Constant (= $12 C_{G1} / u_p d_p$)
	3	CONST3	R	1	Constant (= $12 C_{G2} d_p / \alpha_t u_p$)
	4	CONST4	R	1	Constant (= $1/\sigma_G$)
CCOUNT (SPALDG)					
	1-5	X	R	5	Axial locations (x/d_r) of JETMIX output profiles in current storage
	6-10	NSL	I	5	Number of JETMIX streamlines at JETMIX output stations in current storage
	11	NX	I	1	Number of JETMIX output stations in current storage
	12	XMAX	R	1	Axial location (x/d_r) of last available JETMIX output station
CDXSAV (EMIS)					Used by MFMAIN
	1	DXSAV	R	1	Basic axial step size in jet diameters
CENDS (SPALDG)					
	1	JSTART	I	1	Indicator (in this program, JSTART = 1)
	2	JENDS	I	1	Number of equally spaced at current axial station in finite difference calculation
CEQKIN (LSCK)					General data storage (EQKIN and KINET)
	1	BK	R	1	
	2	AIK	R	1	Not used
	3	DELHNO	R	1	
	4	DEN	R	1	Current value of $(1/m_c - 1/m_u)^{-1}$
	5	XWEQ	R	1	Equilibrium molecular weight
	6	XNO	R	1	Current NO mole fraction
	7	XCO	R	1	Current CO mole fraction

Block Name (Link)	Words	Typical Variable Name	Type	Dim.	Description
CFILK (LMXF, SPALDG)	1	CSC	R	1	Scale parameter for file records (READT, READT2) Scaling constant for file record identification
CGJET (CFILE)	200				Variables stored on various output files but not used in the program reading the file. Used also as dummy or temporary storage input/output operations.
CINIT (SPALDG)	1	XX	R	1	Current axial station (x/d_p) in finite difference calculation
	2	DPSI	R	1	Increment of strain function ($\Delta\psi \sim \text{lbm/ft}^3$) between adjacent equally spaced streamlines at current axial station
CINPJT (EMIS)					Properties of the initial jet extracted by READT2 from the JETMIX output files.
	1	DIAJ	R	1	Jet diameter, inches
	2	MJET		1	Not used
	3	TJET		1	Not used
	4	PTJET		1	Not used
	5	VJET	R	1	Reference velocity of jet, fps
	6	TIJET		1	Not used
	7	PE		1	Not used
	8	VE		1	Not used
	9	ME		1	Not used
	10	TIE		1	Not used
	11	TE		1	Not used
	12-61	X	R	50	Axial stations (in jet diameters) at which JETMIX placed profile data on its output files
	62	GAM		1	Not used
	63	RG		1	Not used
	64	PR		1	Not used
	65	PRT		1	Not used

Block Name (Link)	Words	Typical Variable Name	Type	Dim.	Description
CINPUT (SPALDG)					
	1-500	PSI	R	100,5	Stream function ($\psi \sim \text{lbm/ft}^3$)
	501-1000	R	R	100,5	Radius ($2r/d_p$)
	1001-1500	U	R	100,5	Axial velocity (u/u_p)
	1501-2000	E	R	100,5	Turbulent kinetic energy (e/e_p)
	2001-2500	RHO	R	100,5	Density ($p \sim \text{lbm/ft}^3$)
	2501-3000	XLN	R	100,5	Turbulent length scale ($L_t \sim \text{ft}$)
	3001-3500	F	R	100,5	Fuel concentration ($\text{lb}_m/\text{lb}_m \text{ mixture}$)
CINPUT (EMIS)					
					READT profile properties at five axial stations
	1-5	XX	R	5	Axial location of each station, $\ln(x+1)$
	6-255	RJR	R	5,50	Radius squared (sq. in.)
	256-505	UJR	R	5,50	Velocity, ft/sec
	506-755	RHOJR	R	5,50	Density (not used)
	756-4005	XKJR	R	5,50, 13	Mole fraction of JETMIX dummy gas
	4006-4255	PSIJR	R5,50		Stream function, lb/sec
	4256-4505	CJR	R	5,50	Spalding heterogeneity parameter, $(\text{lb fuel/lb mix})^2$
	4506-4510	NJJ	I	5	Number of profile points
	4511	NII	I	1	= 5
CINSAV (EMIS)					
					GCKP input storage block
	1-57	HGCKP	R	57	Common block used to temporarily save input to GCKP

Block Name (Link)	Words	Typical Variable Name	Type	Dim.	Description
CJETOT (SPALDG)					
	1	GCJ	R	1	Constant ($g_c \times J = 25040$ $\text{lbm-ft}^2/\text{Btu}^c\text{-sec}^2$)
	2	DIAJ	R	1	Initial jet diameter ($d_r \sim \text{in}$)
	3	VJET	R	1	Mass-weighted-average initial jet axial velocity ($u_p \sim$ ft/sec)
	4	EJET	R	1	Initial jet turbulent kinetic- energy ($e_p \sim \text{Btu/lbm}$)
	5	NXTA	I	1	Total number of available JETMIX output stations
CJMIX1 (CFILE, LMXF)	2316				Variables stored on various output files but not used in the program reading the file. Used also as dummy or temp- orary storage during input/ output operations.
CJMIX2 (CFILE, LMXF)	2507				Variables stored on various output files but not used in the program reading the file. Used also as dummy or temp- orary storage during input/ output operations.
CKH02 (LSCK)					Subroutine RATCON - forward rate constants for Hydor- peroxyl reactions
	1	K5	R	1	$\text{H} + \text{O}_2 + \text{M} = \text{HO}_2 + \text{M}$
	2	K6	R	1	$\text{H} + \text{HO}_2 = 2\text{OH}$
	3	K7	R	1	$\text{OH} + \text{HO}_2 = \text{H}_2\text{O} + \text{O}_2$
	4	K8	R	1	$\text{O} + \text{HO}_2 = \text{OH} + \text{O}_2$
	5	K9	R	1	$\text{H} + \text{HO}_2 = \text{H}_2 + \text{O}_2$
CKINET (LSCK)					Main communication common - SCKP/KINET/RATCON
	1	TIME	R	1	Current time into reaction stop, sec
	2	DTIME	R	1	Sub-step size for SCKP reaction, sec

Block Name (Link)	Words	Typical Variable Name	Typical Variable Type	Dim.	Description
CKINET (cont'd.)					
	3	NTSTEP	I	1	Number of intermediate sub- steps for SCKP reaction
	4	XMOLWO	R	1	Not used
	5	CRC	L	1	Not used
	6	NREAC	I	1	Number of reactions 3 body + 2 body
	7-15	RC1	R	9	Arrhenius constants for SCKP reactions
	16-24	RC2	R	9	Temperature exponents for SCKP reactions
	25-33	RC3	R	9	Activation energies for SCKP reactions, cal/gm mole
	34-88	TBE	R	11,5	3 rd body efficiencies for recombination reactions
	89-93	RCON	R	5	Rate constants for 3-body recombination reactions
	94	TK	R	1	Reaction temperature, °K
	95	ENTRY1	L	1	Flag denoting initial entry to KINET T = initial entry
CLOCAL (LMXF)					
					Profile properties of current axial station, interpolated by NEWNET from the five READT profiles
	1-50	RJX	R	50	Radius squared at each READT profile point (sq. in.)
	51-100	UJX	R	50	Velocity at each READT pro- file point (Ft/sec)
	101-150	DUM10		50	Not used
	151-800	KKJX	R	50,13	Mole fraction of each JETMIX dummy gas at each READT profile point
	801	NJ	I	1	Number of READT profile points
CLSPF (JETMIX, LMXF)					
	1	II	I	1	LCFIT, LSPFIT, LFIT Interpolation common Saves interpolation index for use on the next entry

Block Name (Link)	Words	Typical Variable Name	Type	Dim.	Description
CMASS (EMIS)					Mass flows and fuel flows in each NEWNET tube
	1-12	ZMASS	R	12	Total flow in each tube at previous computation station, lbm/sec
	13-24	ZMASSH	R	12	Total flow in each tube at present station, constructed from ZMASS and ALPHA (block CMXFLT)
	25-36	FUELL	R	12	Fuel flow in cold part at present station (only FUELL(1) is used), lbm/sec
	37-48	FUELR	R	12	Fuel flow in hot part at present station (only FUELR(1) is used), lbm/sec
CMATRX (LMXF)					In MXFLUT
	1-12	INTR1		12	Not used
	13	INTR2		1	
	14	DET		1	
	15	IFACTR		1	
CMOLES (LMXF)					Profile data read by READT2 from JETMIX output files
	1-1200	ALX	R	100, 12	Mole fraction of each dummy gas at each point in profile
CMXFLT (EMIS)					Flows of JETMIX dummy gases in NEWNET tubes, and inter-tube mixing parameters
	1-144	W		12,12	W_{ij} is flow of gas i in tube j at previous axial station (lbm/sec)
	145-300	WHAT		13,12	W_{ij} is flow of gas i in tube j at present axial station (lbm/sec) (Last i is mean fuel concentration in tube j)
	301-444	ALPHA		12,12	α_{jk} is fraction of flow in tube j at previous station that is transferred to tube k at present station

Block Name (Link)	Words	Typical Variable Name	Type	Dim.	Description
CNERR (JETMIX)					Constants and error indicator common
	1	BITS	R	1	Junk word used for testing
	2	ERR	L	1	Error indicator (T = error condition)
	3	GC	R	1	$(g_c) 32.174 \text{ ft lb}_m/\text{lb}_f \text{ sec}^2$
	4	GCJ	R	1	$(g_c J) (32.174 * 778) \text{ ft}^2 \text{ lb}_m/\text{Btu sec}^2$
	5	FOOT	R	1	Conversion factor 12 inches/ft
CNFILE (MAIN)					Files present on merge/ decollation file
	1	NF	I	1	1 = PREJET data 2 = PREJET, JETMIX data 3 = PREJET, JETMIX, SPALDG data
CNUNET (SPALDG)					
	1-5	PSJMAX	R	5	Maximum value of stream of function ($\psi \sim \text{lbm}/\text{ft}^2$) at each JETMIX output station in current storage
	6-10	NSLZ	I	5	Number of equally spaced stream- lines at JETMIX output stations in current storage
	11-260	RZ	R	50,5	Radius ($2r/d_p$)
	261-510	UZ	R	50,5	Axial velocity (u/u_p)
	511-760	EZ	R	50,5	Turbulent kinetic energy (e/e_p)
	761- 1010	RHOZ	R	50,5	Density ($\rho \sim \text{lbm}/\text{ft}^3$)
	1011- 1260	XLNZ	R	50,5	Turbulent length scale ($L_t \sim \text{ft}$)
	1261- 1510	FZ	R	50,5	Fuel concentration (lbm/lbm mixture)
					Profiles at the JETMIX output stations in current storage (equally spaced streamlines)

Block Name (Link)	Words	Typical Variable Name	Type	Dim.	Description
CNUNET (EMIS)					Profile properties at current axial station, interpolated by NEWNET from the five READT profiles
	1-50	WFCJX	R	50	Cumulative fuel flow from center of each READT profile print.
	51-100	DUM6		50	Not used
	101-112	WFCUM	R	12	Cumulative fuel flow to outer boundary of each NEWNET tube.
	113-162	DUM7		50	Not used
	163-212	PSIJX	R	50	Stream function (lb/sec) at each READT profile point
COLIMT (LSCK)	1	XCOLIM	R		Limiting (equilibrium) CO mole fraction - used by COCO2B
COND (PREJET, EMIS)					Properties of the reacting gas mixture, used by GCKP
	1	DVAR	R	1	The dependent variable (V=0)
	2	AREA	R	1	Flow area (not used)
	3	MDOT	R	1	Mass flow (not used)
	4	P	R	1	Pressure, atm.
	5	IVAR	R	1	The independent variable (time, sec)
	6	V	R	1	Velocity, cm/sec
	7	RHO	R	1	Density, gm/cm ²
	8	T	R	1	Temperature, K
	9-33	SIGMA	R	25	Gas composition, gmol/gm mix
	34	LS	I	1	Number of species in mixture (= 16)
	35	LSP3	I	1	LS+3
	36	NEXT	L	1	Malfunction has occurred
CONSTF (JETMIX)					Flow scaling (JETMIX)
	1	CON1	R	1	$V_J r^n$ n=1 Plane n=2 Axisymmetric

Block Name (Link)	Words	Typical Variable Name	Type	Dim.	Description
CORED (JETMIX)					Primary jet core data.
	1	XCORE	R	1	Axial coordinate of the point where jet core disappears.
	2	CORE	L	1	T = core has disappeared.
	3	CORED	L	1	T = core disappearance station has been processed.
	4	CORSTP	L	1	T = core has disappeared, do not re-process.
COREQN (EMIS)					Rate constant selection for reaction - $\text{CO} + \text{OH} \rightleftharpoons \text{CO}_2 + \text{H}$.
	1	COREQ	L		T = use special GE rate constants. F = use standard rate constants.
CPARAM (LMXF)					Profile data read by READT2 from JETMIX output files.
	1-100	U	R	100	Velocity, ft/sec.
	101-200	T	R	100	Static temperature, °R.
	201-300	TOT	R	100	Not used.
	301-400	XMACH	R	100	Mach number.
CPBOLI (SPALDG)					
	1-50	G	R	50,1	Mean-square fuel concentration fluctuation intensity (G).
	51	ALPHA	R	1,1	α (coefficient in P.D.E. for determination of G).
	52-101	BETA	R	50,1	β (coefficient in P.D.E. for determination of G).
	102-151	GAMM	R	50,1	γ (coefficient in P.D.E. for determination of G).
	152-201	DELTA	R	50,1	δ (coefficient in P.D.E. for determination of G).
					Profiles at the current axial station in the finite difference calculation of G (equally spaced streamlines).

Block Name (Link)	Words	Typical Variable Name	Type	Dim.	Description
CPCOEF (PREJET)					
	1	CA	R	1	a) Coefficients for the
	2	CB	R	1	b) first NC-1
	3	CC	R	1	c) constituent (inert) gases.
	4	CAA	R	1	a) Coefficients for the
	5	CBA	R	1	b) NC th gas
	6	CCA	R	1	c) (ambient).
					These are coefficients for the determination of constant pressure specific heat ($c_p \sim \text{Btu/lbm-mole } ^\circ\text{R}'$) as a quadratic function of local static temperature ($T \sim ^\circ\text{R}$): $c_p = a + bT + cT^2.$
CPRFL (SPALDG)					
	1-100	PSI	R	100	Stream function ($\psi \sim \text{lbm/ft}^3$).
	101-200	Y	R	100	Radius ($2r/d_p$).
	201-300	UD	R	100	Axial velocity (u/u_p).
	301-400	ED	R	100	Turbulent kinematic energy (e/e_p)
	401-500	RHO	R	100	Density ($\rho \sim \text{lbm/ft}^3$).
	501-600	XLN	R	100	Turbulence length scale ($L_t \sim \text{ft}$).
	601-1800	ALX	R	1200	Molar concentrations of constituent gases (inert species). The variable ALX actually represents a 100 x 12 matrix.
	1801	NPD	I	1	Number of JETMIX streamlines.
					Profiles at an individual JETMIX output station as it is being loaded into current storage.

Block Name (Link)	Words	Typical Variable Name	Type	Dim.	Description
CPRFL (cont'd)					
	1802- 1813	FSPECI	R	12	Fuel concentration in consti- tuent (inert) gases (lbm/lbm mixture).
	1814	NC	I	1	Number of constituent (inert) gases.
	1815- 1864	X	R	50	Axial locations (x/d_r) of available JETMIX output stations.
CPRINT (EMIS)					Diagnostic Print common.
	1-20	PDUM	R	20	Input switches for diagnostic output in mixing and kinetics routines.
CPRJET (CFILE)					
	1387				Variables stored on various output files but not used in the program reading the file. Used also as dummy or temporary storage during input/output operations.
CPROF (LMXF)					Profile data read by READT2 from JETMIX output files.
	1-100	PSI	R	100	Dimensionless stream function.
	101-200	Y	R	100	Dimensionless radius or radius squared.
	201-300	UD	R	100	Dimensionless velocity.
CPROGM (MAIN)					
	1	PROGM	H	1	MAINP - program name current program in execution.
CPROP (JETMIX)					Constant coefficients for mixing scale calculation - (Single jet).
	1-10	CT	R	10	Coefficients of scale calculation.

<u>Block Name (Link)</u>	<u>Words</u>	<u>Typical Variable Name</u>	<u>Type</u>	<u>Dim.</u>	<u>Description</u>
CPROP2 (JETMIX)					Coannular jet - constant coefficients for mixing scale calculation.
	1	CTP	R	1	Scale constant for primary jet.
	2	CTS	R	1	Scale constant for secondary jet.
	3	CTM	R	1	Scale constant for region downstream of merge point.
CPROPJ (LMXF)					Profile data read by READT2 from JETMIX output files (and not used).
	1-100	RHO	R	100	Density.
	101-200	XLN	R	100	Turbulence scale.
CPSEQ2 (LSCK)					
	1	SPM	R	1	
	2	SPMU	R	1	Not used.
	3	SPMC	R	1	
CQIREM (PREJET, LSCK)					Control parameters for QIREM.
	1	YTOL	R	1	Convergence tolerance on difference between Y0 and calculated value of dependent variable.
	2	Y0	R	1	Desired value of dependent variable.
	3	DYDX	R	1	Estimate of first derivative of function.
	4	CTRMAY	R	1	Maximum number of tries before calling ERROR1.
CRATE (LSCK)					
	1	RATE	R	1	Rate of 3-body recombination reactions.
CREACT (EMIS)					In MXFLUT.
	1-12	RHOREA		12	Not used.

Block Name (Link)	Words	Typical Variable Name	Type	Dim.	Description
CREFDA (PREJET)					
	1	DIAJ	R	1	Initial jet diameter ($d_p \sim \text{in.}$).
	2	MJET	R	1	Mass-weighted-average initial jet Mach number.
	3	TIJET	R	1	Initial jet turbulence intensity (u_p'/u_p).
	4	VJFT	R	1	Mass-weighted-average initial jet axial velocity ($u_p \sim \text{ft/sec.}$).
	5-104	GJFT	R	100	Profile at mean-square fuel concentration fluctuation intensity ($G \sim \text{lbm}^2/\text{lbm}^2$) at initial axial station ($\gamma=0$).
	105	PE	R	1	External (ambient) static pressure ($P_e \sim \text{psia.}$).
	106	TE	R	1	External (ambient) static temperature ($T_e \sim ^\circ\text{R.}$).
	107	TTE	R	1	External (ambient) turbulence intensity (u_p'/u_p).
	108	VE	R	1	External (ambient) axial velocity ($u_e \sim \text{fps.}$).
	109	GEX	R	1	External (ambient) mean-square fuel concentration fluctuation intensity ($G_{ex} \sim \text{lbm}^2/\text{lbm}^2$).
	110	RG	R	1	Gas constant ($\text{ft-lbf/lbm-}^\circ\text{R.}$).
	111	PR	R	1	Prandtl number.
	112	PRT	R	1	"Turbulent" Prandtl number.
	113	SC	R	1	Sutherland constant for viscosity calculation ($S_c \sim ^\circ\text{R.}$).
	114	TREF	R	1	Reference temperature for viscosity calculation ($T_{ref} \sim ^\circ\text{R.}$).
	115	MUREF	R	1	Reference viscosity for viscosity calculation ($\mu_{ref} \sim \text{lbm/ft-sec.}$).

Block Name (Link)	Words	Typical Variable Name	Type	Dim.	Description
CREFDA (cont'd.)					
116		DIFF	L	1	T = species diffusion calculated, F = species diffusion not calculated.
117		NC	I	1	Number of constituent gases (inert species) in jet and external flow.
118-129		CNAME	R	12	Names of constituent gases.
130-141		ALE	R	12	External (ambient) molar concentrations of constituent gases.
142-153		SCM	R	12	Effective Schmidt numbers of constituent gases.
154-189		CPC	R	36	Coefficients in polynomial representation of molar heat capacities of constituent gases ($a_1, b_1, c_1, a_2, \dots$): $C_{pi} = a_i + b_i T + c_i T^2$.
190		NJ	I	1	Number of JETMIX streamlines in jet at initial axial station ($\gamma = 0$)
191		NM	I	1	Maximum number of JETMIX streamlines before mesh re- distribution occurs.
192		CT1	R	1	C_{t1} } Constants in the empirical equation for turbulence length scale in the potential core region.
193		CT2	R	1	
194		CT3	R	1	C_{t3} } Constants in the empirical equation for turbulence length scale in the transition region.
195		CT4	R	1	
196		CT5	R	1	
197		CT6	R	1	
198		CT7	R	1	
199		CT8	R	1	C_{t8} Constant in the empirical equation for turbulence length scale in the fully developed region.

Block Name (Link)	Words	Typical Variable Name	Type	Dim.	Description
CREFDA (cont'd.)					
	200-299	Y	R	100	Radius ($2r/d_p$).
	300-399	UD	R	100	Axial velocity ($u \sim \text{ft/sec}$).
	400-499	THD	R	100	Static temperature ($T \sim ^\circ\text{R}$).
	500-599	TID	R	100	Turbulence intensity (u'/u_p).
	600-1799	ALX	R	1200	Molar concentrations of constituent gases. (Actually represents a 100 x 12 matrix)
CSLDAT (SPALDG)					
	1-50	SY	R	50	Stream function ($\psi \sim \text{lbm/ft}^3$).
	51-100	RAD	R	50	Radius ($2r/d_p$).
	101-150	VEL	R	50	Axial velocity (u/u_p).
	151-200	TKE	R	50	Turbulent kinetic energy (e/e_p).
	201-250	DEN	R	50	Density ($\rho \sim \text{lbm/ft}^3$).
	251-300	TLS	R	50	Turbulence length scale ($L_t \sim \text{ft}$).
	301-350	FAR	R	50	Fuel concentration (lbm/lbm mixture).
CSFALD (CFILE)					
	51				Variables stored on various output files but not used in the program or temporary storage during input/output operations.

Profiles at the initial axial station (JETMIX streamlines).

Profiles at the current axial station in the finite difference calculation of G.

Block Name (Link)	Words	Typical Variable Name	Type	Dim.	Description
CSPARE (LMXF)					Temporary storage for redistributed JETMIX and SPALDG profile points in READT2.
	1-1600	H	R	1600	Temporary values of Y^2 , U, ρ , ALX, G.
	1601-1650	PSIR	R	50	Values of ψ/ψ_c generated by NEWPSI
	1651-1700	G	R	50	Values of G, ψ , and radius read from SPALDG profile data output files.
	1701-1750	SY	R	50	
	1751-1800	RAD	R	50	
CSPECI (EMIS)					Miscellaneous parameters used by MXFLUT, NEWNET, etc.
	1	NSPECI	I	1	Number of JETMIX dummy gases (includes ambient air).
	2	NF	I	1	Storage location for fuel flow (= NSPECI + 1).
	3	DX	R	1	Current axial step size in jet diameters (from block CINPJT).
CSTART (SPALDG)					
	1-100	GJET	R	100	Profile of mean-square fuel concentration fluctuation intensity(G) at initial axial station (x=0).
	101	GEX	R	1	External (ambient) value of mean-square fuel concentration fluctuation intensity (G).
CTABPR (MAIN)					Subroutine TABPRT.
	1	ITAB	I	1	Sets array index for starting location of printing.
CTHETA (SPALDG)					
	1	THETA	R	1	Crank-Nicolson factor for finite difference solution of the parabolic partial differential equation for G (THETA = 0 ~ explicit solution, THETA = 1/2 ~ Crank-Nicolson, THETA = 1 ~ fully implicit solution.

Block Name (Link)	Words	Typical Variable Name	Type	Dim.	Description
CTHETA (cont'd.)					
	2	II	I	1	Indicator (in this program, II = 1).
CTRANS (SPALDG)					
	1-250	RZT	R	5,50	Transpose of RZ
	251-500	UZT	R	5,50	Transpose of UZ
	501-750	EZT	R	5,50	Transpose of EZ
	751-1000	RHOZT	R	5,50	Transpose of RHOZ
	1001-1250	XLNZT	R	5,50	Transpose of XLNZ
	1251-1500	FZT	R	5,50	Transpose of FZ
CTRIDI (SPALDG)					
	1-150	A	R	50,3	Tridiagonal coefficient matrix
	151-200	B	R	50	"right-hand side" vector.
CTRL (JETMIX)					
					Calculation control.
	1	NXTA	I	1	Number of main calculation stations.
	2	CMPRS	L	1	F = incompressible. T = compressible.
	3	QJET	L	1	F = isothermal jet. T = non-isothermal jet.
	4	TURBJ	L	1	F = laminar jet. T = turbulent jet.
	5-14	COEF	R	10	Constant quantities used in the calculation of the difference equation coefficients

<u>Block Name (Link)</u>	<u>Words</u>	<u>Typical Variable Name</u>	<u>Type</u>	<u>Dim.</u>	<u>Description</u>
CTRL (cont'd.)					
	15	NPU	I	1	Number of upstream profile points.
	16	NPD	I	1	Number of downstream profile points.
	17	DXC	R	1	Current ΔX to next station.
	18	XU	R	1	Upstream X location.
	19	XD	R	1	Downstream X location.
	20-819	DSTOR	R	800	Dummy - scratch storage.
CTRL2 (JETMIX)					
					Control storage - coannular jet.
	1	EDGEI	R	1	Y location of edge of primary jet.
	2	SFI	R	1	ψ (streamfunction) value at EDGEI.
	3	MERGE	L	1	F = coannular jets have not merged. T = coannular jets have merged.
	4	XMERGE	R	1	X location of merge points of coannular jets.
	5	YMERGE	R	1	Y or R location of merge point of coannular jets.
	6	SLOPEI	R	1	Slope of straight line from primary nozzle lip to merge point.
	7	SLOPEO	R	1	Slope of straight line from secondary nozzle lip to merge point.
	8	CEPTI	R	1	Intercept of straight line from primary nozzle lip to merge point.
	9	CEPTO	R	1	Intercept of straight line from secondary nozzle lip to merge point.
CXLOCA (EMIS)					
					Used by READT2.
	1-5	XX	R	5	Five axial distances in jet diameters for which READT2 has stored JETMIX profile data.

<u>Block Name (Link)</u>	<u>Words</u>	<u>Typical Variable Name</u>	<u>Type</u>	<u>Dim.</u>	<u>Description</u>
CYOLD (LMXF)	1-12	YRICH0		12	In MXFLUT. Not used.
DFIT (JETMIX)	1-100	CLSP	R	100	Confined jet - curve fit. Coefficients for LSP curve fit of outer wall contour.
DICTRL (JETMIX)	1	DIFF	L	1	Control storage - species diffusion equations. T = solve species diffusion equations. F = Do not solve species diffusion equations.
	2-11	CND	R	10	Constant quantities used in the calculation of the diffusion difference equation coefficients.
DIFEQI (JETMIX)	1	NC	I	1	Species diffusion - input common. Number of species - maximum = 6.
	2-7	CNAME	H	6	Species names.
	8-13	ALJ	R	6	Primary jet - reference species mole fractions
	14-19	ALJO	R	6	Secondary jet - reference species mole fraction.
	20-25	SCM	R	6	Species Schmidt Numbers.
	26-31	TCPRF	R	6	Not used.
	32-37	HCPRF	R	6	Not used.
	38-55	CPC	R	3,6	Species molar heat capacity. Coefficients of parabolic fit vs. absolute temperature.
DPBETA (LSCK)	1	BETADP	D	2	Double precision value of β - used by PSEQ2

Block Name (Link)	Words	Typical Variable Name	Type	Dim.	Description
DQIREM (PREJET, EMIS)	1	KINFIL	I	1	Kinetics/Equilibrium T calc. scratch file. File code for output of SCKP initial conditions/Equilibrium T calculation iteration history to be dumped by call to ERROR1.
EDGE (JETMIX)	1	YEDGE	R	1	Coordinate of jet edge. Normal coordinate (Y) at jet edge.
	2	SFEDGE	R	1	Streamfunction (ψ) at jet edge.
ERASE (LSK, PREJET)	36				Scratch common used by QIREM iteration routine.
FILK (JETMIX)	1	CSC	R	1	Scale parameter for file records. Scaling constant for file record identification.
FLOBAL (JETMIX)	1	MAXIT	I	1	Confined jet - parameters for pressure - area iteration. Maximum allowable P-A iterations.
	2	SUPB	L	1	T = select supersonic branch during P-A iteration. F = select subsonic branch.
	3	PSID	R	1	Streamfunction at duct outer wall.
	4	YDD	R	1	Local duct outer wall Y.
	5	YDC	R	1	Local center body Y.
	6	P1	R	1	Upstream static pressure.
	7	P2	R	1	Downstream static pressure.
	8	UCL	R	1	Local u or centerbody u/u_j .
	9	TOL	R	1	Iteration tolerance - P-A iteration.
	10	UPSTRM	L	1	T = upstream of intersection of mixing jet and duct wall. F = downstream of intersection of mixing jet and duct wall.
	11	CVG	L	1	T = P-A iteration has converged.

<u>Block Name (Link)</u>	<u>Words</u>	<u>Typical Variable Name</u>	<u>Type</u>	<u>Dim.</u>	<u>Description</u>
GASCMF (PREJET, EMIS)					Detailed properties of hot and cold gas parts of each tube flow at previous and present computation station (see Table 2 for subscript key).
	1-24	YY	R	12,2	Mass fraction hot part of gas (subscripts K,L).
	25-72	FUEL	R	2,12, 2	Fuel concentration, lb fuel/lb mix (subscripts J,K,L).
	73-120	ENTH	R	2,12, 2	Absolute enthalpy, Btu/lbm (subscripts J,K,L).
	121-888	CONC	R	16,2, 12,2,	Species concentrations, miles I/lb mix (subscripts I,J,K,L).
	889-912	HCINCP	R	12,2	"Reactive incipency" of hydrocarbons in hot part (subscripts K,L).
	913-936	U	R	12,2	Velocity, ft/sec (subscripts K,L).
	937-1104	OTHER		168	Unused space.
GASTMW (EMIS)					Communication between mixing/homogenization and kinetics routines (see Table 2 for subscript key).
	1-48	TG	R	2,12, 2	Static temperature, °R [TG(J,K,L)].
	49-96	MWTG	R	2,12, 2	Molecular weight, lbm/lb mole [MWTG(J,K,L)].
	97-120	TAU	R	12,2	Reaction time for a [TAU (K,L)] given streamtube.
	121-168	CPG	R	2,12, 2	Constant pressure heat capacity, Btu/lbm °R [CPG(J,K,L)].
GHSC (PREJET, EMIS)					Thermo properties of pure chemical species.
	1-25	GRT	R	25	Dimensionless Gibbs free energy G/RT.
	26-50	HRT	R	25	Dimensionless enthalpy H/RT.
	51-75	SR	R	25	Dimensionless entropy S/R.

Block Name (Link)	Words	Typical Variable Name	Type	Dim.	Description
GHSC (cont'd.)					
	76-100	CPR	R	25	Dimensionless heat capacity C_p/R .
	101-125	DCPR	R	25	$(dC_p/dT)/R$.
INDATA (PREJET, EMIS)					
	1	N,HC	R	1	Fuel hydrogen/carbon atom ratio.
	2	HF	R	1	Absolute enthalpy of fuel, Btu/lbm.
	3	WAR	R	1	Water/air mass ratio in ambient air.
	4	T2	R	1	Fan inlet total temperature, °R.
	5	BETA	R	1	Engine by-pass ratio.
	6	T25	R	1	Fan discharge total temperature, °R.
	7	FAR5	R	1	Turbine discharge fuel/air mass ratio.
	8	EINO2C	R	1	NO_x emission index of main combustor, lb NO_2 /lb fuel.
	9	PO	R	1	Ambient static pressure, psia.
	10-20	RCO	R	11	Point sample measurements of CO and CO_2 , mole fraction of sample dried to 32°F saturation.
	21-31	RCO2	R	11	
	32-42	RHC	R	11	Sample measurements of hydrocarbon (as CH_4) and total oxides of nitrogen, mole fraction in net sample.
	43-53	RNOX	R	11	
	54-64	PT	R	11	Probe impact pressure measurement, psia.
	65-75	PS	R	11	Local static pressure, psia.
	76-86	BLOC	R	11	Local effective by-pass ratio.
	87-97	EICOC	R	11	CO emission index of main combustor, lb CO/lb fuel.
INMOLF (LSCK)					
	1-12	XIN	R	1	Initial mole fractions at the beginning of a kinetics step, mols i/mole mix.

<u>Block Name (Link)</u>	<u>Words</u>	<u>Typical Variable Name</u>	<u>Type</u>	<u>Dim.</u>	<u>Description</u>
INNAME (JETMIX)					Column vector names - JETMIX input (not used)
	1	NB	I	1	Number of B-input columns.
	2-6	TAB	H	5	Names of input columns.
	7	ND	I	1	Number of D-input columns.
	8-11	TAD	H	4	Names of D-input columns.
INP1 (JETMIX)					Entry control block (JTSTEP).
	1	ENTRY1	L	1	Set T for first entry into JTSTEP.
INPJET (JETMIX)					Jet input parameters
	1	DIAJ	R	1	Primary jet diameter, inches or width.
	2	MJET	R	1	Primary jet reference Mach number.
	3	TJET	R	1	Primary jet reference static temperature, °R.
	4	PTJET	R	1	Primary jet reference stagna- tion pressure, psia.
	5	VJET	R	1	Primary jet reference velocity, fps.
	6	TIJET	R	1	Primary jet turbulence in- tensity (relative to VJET).
	7	PE	R	1	Boundary condition - static pressure, psia.
	8	VE	R	1	Boundary condition - external velocity, fps.
	9	ME	R	1	Boundary condition - external Mach number.
	10	TIE	R	1	Boundary condition - turbulence intensity
	11	TE	R	1	Boundary condition - static temperature, °R.
	12	AXI	L	1	T = axisymmetric jet. F = plane or 2-D jet.
	13	NJ	I	1	Index of nozzle lip mesh point.

Block Name (Link)	Words	Typical Variable Name	Type	Dim.	Description
INPJET (cont'd.)					
	14	NMAX	I	1	Number of mesh points allowed before grid is refined.
	15-114	X	R	100	Table of primary X/D values.
	115-214	XPRN	L	100	Reference - X Table of print stations. T = Print corresponding X station.
	215	GAM	R	1	Not used.
	216	RG	R	1	Gas constant ft lbf/sec ² °R.
	217	PR	R	1	Prandtl number $C_p \mu/K$.
	218	PRT	R	1	Turbulent Prandtl number, $C_p \mu_t/K_t$.
	219	SC	R	1	Sutherland constant (viscosity formulation).
	220	TREF	R	1	Reference temperature for viscosity calculation.
	221	MUREF	R	1	Reference viscosity at TREF.
IOFILE (MAIN)					I/O file indication.
	1	TAPIN	L	1	T = input file. T = output file.
JET (JETMIX)					Properties of jet along axis of symmetry or on centerbody.
	1-100	B	R	100	Non-dimensional width of mixing zone.
	101-200	UC	R	100	Velocity (u/u_j).
	201-300	TC	R	100	Static temperature (T/T_j).
	301-400	TIC	R	100	Turbulence intensity (u').
	401-500	PTC	R	100	Stagnation pressure ratio $\frac{(P_T - P_E)}{(P_{Tjet} - P_E)}$
	501-600	WJ	R	100	Relative entrained flow W/W_j .
	601-700	YJ	R	100	Y location of jet edge.

Block Name (Link)	Words	Typical Variable Name	Type	Dim.	Description
JET1 (JETMIX)					Jet parameters.
	1	EJET	R	1	Reference turbulent kinetic energy for primary jet, Btu/lbm.
	2	FLOWJ	R	1	Normalizing jet flow rate at initial station.
	3	NXN	I	1	Current calculation station index.
	4	TTO	R	1	Jet stagnation temperature, °R.
JET2 (JETMIX)					Station stagnation temperature.
	1-100	TTC	R	100	Stagnation temperature ratio on jet axis or center body. $\frac{T_{Tc} - T_e}{T_{To} - T_e}$
JET3 (JETMIX)					Control common for added stations. Confined mixing.
	1	NV	I	1	Added station counter.
	2	STADD	L	1	T = skip property calculation at added station.
	3	STATF	L	1	T = do not add to "additional station counter" (NV)
JETDAT (PREJET, EMIS)					Mean properties of flow at nozzle exit survey points. Also info transmitted from MXFLUT to HYCARB.
	1	NPTS	I	1	Number of survey points (eleven max.; point NPTS+1 is ambient air).
	2-13	RAD	R	12	Radial location, ft.
	14-25	TS	R	12	Static temperature, °R.
	26-37	U	R	12	Velocity, fps.
	38-49	SPV	R	12	Specific volume, ft ³ /lbm.
	50-61	MWT	R	12	Mean molecular weight, lbm/mole.
	62-73	CP	R	12	Specific heat, Btu/lbm °R.

Block Name (Link)	Words	Typical Variable Name	Type	Dim.	Description
JETDAT (cont'd.)					
	74-85	FUEL	R	12	Fuel concentration, lb fuel/ lb mix
	86-97	SPALDG	R	12	Heterogeneity parameter, (lb fuel/lb mix) ² .
	98-109	TKE		12	Not used.
	110-121	TFR	R	12	Mass of fuel transported into the hot part of a tube from the hot parts of all tubes in a mixing step.
	122	TFL	R	1	Mass of fuel transported into the hot part of a tube from the cold part of that tube in a homogenization step.
	123-134	TIM1	R	12	Cumulative residence time of all tubes at previous compu- tation station.
	135	TIM2	R	1	Cumulative residence time of a tube at present computation station.
	136-145	OTHER		10	Unused space.
JETTWO (JETMIX)					Coannular jet input parameters.
	1	TWO	L	1	T = Coannular jet. F = Single jet.
	2	DIAO	R	1	Secondary jet diameter, inches or width.
	3	MJETO	R	1	Secondary jet reference Mach number.
	4	TJETO	R	1	Secondary jet reference static temperature, °R.
	5	VJETO	R	1	Secondary jet reference velocity, fps.
	6	PTJETO	R	1	Secondary jet reference stag- nation pressure, psia.
	7	TIJETO	R	1	Secondary jet turbulence intensity (relative to VJET).
	8	NJO	I	1	Index of secondary nozzle lip mesh point.

Block Name (Link)	Words	Typical Variable Name	Type	Dim.	Description
JMIXIC (LMXF)	2199				Variables stored on various output files but not used in the program reading the file. Used also as dummy or temporary storage during input/output operations.
KEYS (GMOD, JETMIX, LMXF, PREJET, SPALDG)					File record indices.
	1-10	KEYA	I	10	File record indices for identification.
	11-20	KEYB	I	10	Not used.
	21-30	RODA	I	10	Not used.
	31-40	RODB	I	10	Not used.
KININS (LSCK)					Data storage for KINET.
	1	XMWUN	R	1	Unreacted molecular weight of fuel-air mixture, lbm/lb mole.
	2	XMWC	R	1	Fully recombined molecular weight of combustion products, lbm/lb mole.
	3	TCONST	L	1	Not used. Set F.
	4	CONER	L	1	Not used. Set T.
	5	XNOI	R	1	Initial mole.
KOUT (EMIS)					Information used by GCKP (see Reference 3).
	1-20	TITLE	H	20	Page heading text, same as in OPCTRL
	21	UNITI	H	1	Input units (= FPS)
	22	UNITO	H	1	Output units (= FPS)
	23	CONC	L	1	Output composition will be converted to mole fractions (= .TRUE.).
	24	EXCHR	L	1	Energy exchange rates rather than conversion rates will be output (= .FALSE.).
	25-54	DELH	R	30	Heat of reaction.
	55	FPS	H	1	"FPS"
	56	SI	H	1	"SI"
	57	DBUGO	L	1	Print intermediate output (= .FALSE., change via Name-list PROB).

<u>Block Name (Link)</u>	<u>Words</u>	<u>Typical Variable Name</u>	<u>Type</u>	<u>Dim.</u>	<u>Description</u>
LLTERP (JETMIX)					Transition region scale selection.
	1	LTERP	L	1	T = linear variation of turbulence scale. F = Exponential variation of turbulence scale.
LTUS (LGCK)					File codes used in NASA GCKP program (Reference 3). Not used in PLUMOD.
	1	LTHM	I	1	File containing polynomial coefficients for species thermo properties.
	2	LDAT	I	1	Scratch file for storing input data.
MERGET (JETMIX)					Coannular jet - Merge information.
	1	MER	L	1	Merge indicator - Set to T on merge. Set to F after processing merge station.
	2	MERSTP	L	1	Merge indicator - Set to T on merge.
	3	XMRG	R	1	X location of jet merge
MISC (JETMIX)					Diagnostic print block.
	1-20	PM	R	20	Diagnostic print indicators.
MIXER (JETMIX)					Confined jet - input parameters.
	1	MIX	L	1	T = confined mixing. F = free mixing
	2-101	RDD	R	100	Outer duct radius or Y location, inches.
	102-201	XD	R	100	X station location, inches.
	202	CF	R	1	Not used.
	203	YR	R	1	RD nondimensionalized by DIAJ/2.
MIXPRP (JETMIX)					Confined mixing - inviscid stream parameters (not used).
	1-100	MA2	R	100	Mach number.
	101-200	VE2	R	100	Velocity, fps.

Block Name (Link)	Words	Typical Variable Name	Type	Dim.	Description
MIXPRP (cont'd.)					
	201-300	TE2	R	100	Static temperature, °R.
	301-400	TWC	R	100	Wall temperature, °R.
MOLES (JETMIX)					
	1-1200	ALJ	R	100, 12	Species mole fraction storage (working). Species mole fractions i = 1, 2.
MOLUP (CPLINK)					
	1-1200	ALXU	R	100, 12	Species mole fraction storage (working) TKE working storage. Species mole fractions at upstream station.
	1201-2400	DALXU	R	100, 12	Normal deviative of species mole fractions.
	2401-2600	DTKE	R	200	Normal deviative of turbulent kinetic energy.
NOXTRA (LSCK)					
	1	DNOXDT	R	1	Subroutine NOX2B Rate of NOX production/consump- tion $\frac{dNO}{dt} \times$.
OPCTRL (PREJET, EMIS)					
	1-20	TITLE	H	20	Output control information. Six-character Hollerith words transmitted to page headings.
	21-50	PRINT	R	30	Axial stations (feet) at which emissions summation and print-out are desired.
OPTS (LGCK)					
	1	VERSI	H	1	GCKP input options (Reference 3). See NAMBLK. "TIME", the independent variable.
	2	TIMEV	H	1	"TIME"
	3	VERSA	H	1	"AREA", the assigned variable.
	4	AREAV	H	1	"AREA"
	5	ELIM	L	1	Automatically eliminate from error consideration species with nonrepresentative errors.
	6	TCON	L	1	Hold temperature constant.
	7	RHOCON	L	1	Hold density constant.
	8	IPRCOD	I	1	Input type indicator (not used).

Block Name (Link)	Words	Typical Variable Name	Type	Dim.	Description
OUTMIX (JETMIX)					Output common block.
	1	NXORIG	I	1	Number of calculation stations.
PARAM (JETMIX)					Communication common - Co-efficient of different equations (DFEQ).
	1-200	AL	R	200	Coefficient α .
	201-400	BE	R	200	Coefficient β .
	401-600	GM	R	200	Coefficient γ .
	601-800	EPS	R	200	Coefficient e .
	801-1000	DL	R	200	Coefficient δ .
	1001-1200	VAR	R	200	Dependent variable $\begin{Bmatrix} U \\ \theta \\ E \\ K_1 \end{Bmatrix}$.
	1201-1400	SM1	R	200	Independent variable (ψ) at upstream station.
	1401	NM1	I	1	Number of mesh points at upstream station.
	1402-1601	SM	R	200	Independent variable (ψ) at downstream (current) station.
	1602	NM	I	1	Number of mesh points at downstream station.
	1603	DX	R	1	Streamwise step size (ΔX).
	1604	B1	R	1	Boundary condition coefficients applied at mesh point 1.
	1605	C11		1	
	1606	D1		1	
	1607	AN	R	1	Boundary condition coefficients applied at mesh point NM.
	1608	BN		1	
	1609	CN		1	
PARAM1 (CPLINK)					Communication common - coefficients of difference equations (DFEQ)
	1-200	ETA	R	200	Coefficient η .

AD-A045 627

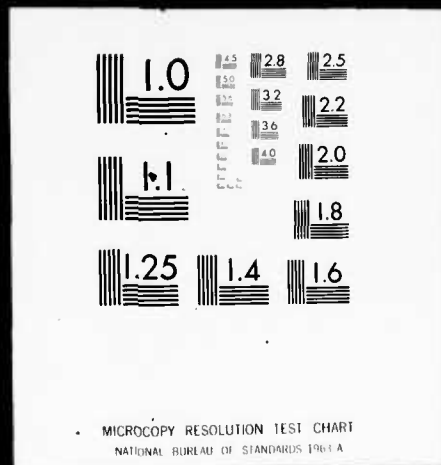
GENERAL ELECTRIC CO CINCINNATI OHIO AIRCRAFT ENGINE GROUP F/G 21/5
DEVELOPMENT OF EMISSIONS MEASUREMENT TECHNIQUES FOR AFTERBURNIN--ETC(U)
OCT 75 W C COLLEY, D R FERGUSON, M A SMITH F33615-73-C-2047
R75AEG459 AFAPL-TR-75-52-SUPPL-2 NL

UNCLASSIFIED

2 of 5
ADA045 627



2 OF 3
ADA045 627



Block Name (Link)	Words	Typical Variable Name	Type	Dim.	Description
PORIDE (EMIS)					Input storage to override print stations set in PRELIM overlay.
	1-30	PSTA	R	30	Print station over-ride. Values will be used in deference to those in OPCTRL (jet diameters).
PQRE (LGCK)					Internal storage for GCKP (reference 3).
	1-28	PK	?	28	
	29-56	QK	?	28	
	57-84	RK	?	28	
	85-112	E	?	28	
PRIN (LGCK)					Output control for NASA GCKP (reference 3).
	1-50	PRINT	R	50	Values of independent variable at which output is desired.
	51	NPRNTS	I	1	Number of values of PRINT.
	52	END	R	1	Last value of independent variable at which output is desired.
	53	EVSTEP	L	1	Print after every integration step.
PROF (JETMIX)					Profiles across jet.
	1-200	PSI	R	200	Stream function (ψ).
	201-400	Y	R	200	Normal coordinate $Y = 2Y/DJ$.
	401-600	UD	R	200	Velocity ($U = u/u_J$).
	601-800	THD	R	200	Static temperature ($\theta = T/T_J$).
	801-1000	ED	R	200	Turbulent kinetic energy ($E = e/e_J$).
PROPJ2 (JETMIX)					Coannular jet - properties.
	1	MACHO	R	1	Reference Mach number for scale calculation.
	2	REFLO	R	1	Reference scale for secondary jet.

<u>Block Name (Link)</u>	<u>Words</u>	<u>Typical Variable Name</u>	<u>Type</u>	<u>Dim.</u>	<u>Description</u>
PROPJ2 (cont'd.)					
	3	YI	R	1	Y coordinate - edge of mixing zone of primary jet.
	4	YO	R	1	Y coordinate - edge of mixing zone of secondary jet.
	5	MERGE	L	1	T = downstream of merge point.
PROPJT (JETMIX)					
					Transport and thermodynamic properties.
	1	P	R	1	Static pressure, psia.
	2	PRL	R	1	Prandtl number, $C_p \mu / K$.
	3	PRT	R	1	Turbulent Prandtl number, $C_p \mu_t / K_t$.
	4	RGAS	R	1	Gas constant RG.
	5	SC	R	1	Sutherland constant, °R.
	6	TREF	R	1	Reference temperature for viscosity calculation, °R.
	7	MUREF	R	1	Viscosity at TREF, lbm/ft sec.
	8	MACH	R	1	Reference Mach number.
	9	XLC	R	1	X location of disappearance of jet potential core.
	10	REFL	R	1	Reference scale of turbulence.
	11	C	R	1	Constant for turbulent viscosity calculation.
	12	CHI	R	1	Constant for turbulent viscosity calculation.
	13	RNORM	R	1	Normalizing Reynold's number of turbulence
	14-213	RHO	R	200	Density, lbm/ft ³ .
	214-413	MUL	R	200	Laminar dynamic viscosity (μ) lbm/ft sec.
	414-613	KCP	R	200	Ratio of thermal conductivity to heat capacity, K/C_p .
	614-813	MUEFF	R	200	Turbulent + laminar viscosity ($\mu + \mu_t$).

Block Name (Link)	Words	Typical Variable Name	Type	Dim.	Description
PROPJT (cont'd.)					
	814- 1013	XLN	R	200	Turbulence scale.
	1014- 1213	DK	R	200	Dissipation (turbulence) coefficient.
	1214- 1413	RETURB	R	200	Reynold's number of turbulence $\frac{\rho v_e L_t}{\mu}$
PROPR (LSCK)					
					Subroutine KINET - thermodynamic quantities (equilibrium) (not used).
	1	PR	R	1	Pressure.
	2	HHR	R	1	Enthalpy.
	3	TR	R	1	Temperature.
	4	FCR	R	1	Fuel/air ratio.
	5	RHOR	R	1	Density.
	6	RR	R	1	Gas constant.
	7	WMTR	R	1	Molecular weight.
PSEQ (LSCK)					
					SCKP - data storage. Com- munication common between kinetics routines.
	1	FOA	R	1	Fuel/air ratio, lbm/lbm.
	2	BETS	R	1	Combustion efficiency, (β).
	3	TP	R	1	Reaction temperature, °R.
	4-19	X	R	16	Constituent mole fractions, mols i/mole mix.
	20	DHQDMW	R	1	Not used.
	21	TEQ	R	1	Equilibrium reaction temper- ature, °R.
	22	BEQ	R	1	Equilibrium combustion efficiency, (β_{eq}).
	23	XMWT	R	1	Mixture molecular weight, lbm/lb mole.
	24	HNEQ	R	1	Not used.

Block Name (Link)	Words	Typical Variable Name	Type	Dim.	Description
PSEQX (LSCK)					Fuel and air communication common
	1	HQC	R	1	Fuel H/C ratio.
	2	HUM	R	1	Air humidity, lbm H ₂ O/lbm dry air
	3	CO2AIR	R	1	Air - CO ₂ concentration, mol CO ₂ /mol air.
	4	MAIR	R	1	Air molecular weight, lbm/lb mole
	5	FS	R	1	Stoichiometric fuel/air ratio, lbm/lbm.
	6	FUELMW	R	1	Fuel molecular weight, lbm/lb mol 1gm atom of C - basis.
RATIO (JETMIX)					
	1	AMBT0	L	1	T = ambient stagnation temperature set. (JETMIX)
REAC (LGCK)					Chemical reaction information used by GCKP (Reference 3).
	1-120	LSR	I	4,30	Code number of species involved in each reaction (see BLKRDA).
	121-150	XX	?	30	Internal GCKP variable.
	151-180	RATE	?	30	"
	181-210	LKEQ	?	30	"
	211-240	DLKEQ	?	30	"
	241-270	MM	?	30	"
	271	LR	?	1	"
RRAT (LGCK)					Reaction rate coefficients and third-body efficiencies for GCKP (Reference 2).
	1-30	A	R	30	Coefficients for Arrhenius expressions for reaction rates Rate = AT ^N exp (-E/RT).
	31-60	N	R	30	
	61-90	EACT	R	30	
	91-120	B	?	30	Internal GCKP variable.
	121-870	M	R	25,30	Third-body efficiencies.
	871	ALLM1	L	1	All third-body efficiencies are 1.0.

Block Name (Link)	Words	Typical Variable Name	Type	Dim.	Description
RSTART (JETMIX)					Calculation restart parameters.
	1	MIXPRE		1	Not used.
	2	NREG	I	1	1 = potential core region. 2 = transition region. 3 = similar region.
	3	NRES	I	1	Restart (X) station.
	4	RESTRT	L	1	T = restart case.
SCALED (JETMIX)					Turbulence scale from species diffusion equation.
	1	SCLD	L	1	T = use mole fraction of ambient species for determin- ation of mixing zone width reference dimension. F = use velocity profile.
	2	ALXLIM	R	1	Limiting value for detection of inner edge of mixing zone in the "potential core region".
SCALER (JETMIX)					JETMIX input scalars.
	1	SP	R	1	Pressure.
	2	SV	R	1	Velocity
	3	SLEN	R	1	Length.
SETNEW (JETMIX)					JETMIX Jet edge common.
	1	LKK	I	1	Mesh index - ≤ 3 forces addition of mesh point at jet edge.
	2	LCOR	I	1	Not used.
SING (JETMIX)					Dummy JETMIX input common.
	1-43	SSD	R	43	Temporary input storage for single cell input.

Block Name (Link)	Words	Typical Variable Name	Type	Dim.	Description
SINT (LGCK)					Integration step size controls for GCKP (Reference 3).
	1	HMIN	R	1	Minimum step size, sec.
	2	HINT	R	1	Initial step size.
	3	HN	R	1	Last step size.
	4	HNP1	R	1	Next step size.
	5	HMAX	R	1	Maximum step size.
	6	NH	I	1	Number of steps since last print.
	7	AVH	R	1	Average step size since last print.
	8	EMAX	R	1	Maximum tolerable integration error.
	9	ERRN	R	1	Estimated error from proposed next step size.
	10	JCV	?	1	GCKP internal variable.
	11	KOUNT	?	1	"
	12	ERRP	?	.	"
SNMW (EMIS)					Chemical species names and molecular weights (see BLCK).
	1-150	ALSP	H	2,75	Specie names, 12 characters, left-justified.
	151-225	ALMW	R	75	Specie molecular weights ($M_o \equiv 16$).
SPEC (LGCK)					Miscellaneous chemical reaction info used by GCKP (Reference 2).
	1-60	SNAM	H	2,30	Names of chemical species copied from block SNMW starting at SNAM (1,4). See also NAMBLK.
	61-85	MW	R	25	Specie molecular weights, copied from block SNMW.
	86-110	W	?	25	Internal GCKP variable.
	111-860	STOIC	R	25,30	Stoichiometric coefficients for reactions (see BLKST).
	861- 1610	OMEGA	?	25,30	Internal GCKP variable.

Block Name (Link)	Words	Typical Variable Name	Type	Dim.	Description
STA2 (JETMIX)					Confined jet - iterated conditions at downstream station.
	1	MACH2	R	1	Mach number.
	2	TS2	R	1	Static temperature, °R.
		SS2	R		Sonic velocity, fps.
	3	V2	R	1	Velocity, fps.
	4	RHO2	R	1	Density, lbm/ft ³ .
	5	DPDX2	R	1	Pressure gradient, psi/in (dP/dx).
STCTRL (EMIS)					Control information for mixing/homogenization and kinetics computations.
	1	LSTA	I	1	Current calculation station index.
	2	FINAL	L		T = calculation has reached a streamwise print station.
	3	CHEMK	I	1	Chemical kinetics selection switch 1 = GCKP 2 = SCKP
	4	FIRSTM	L	1	Indicator for initial mixing step. Set T on first entry to MXFLUT. F thereafter.
	5	FIRSTC			Not used.
	6	SC			Not used.
	7	DXC	R	1	Not used.
	8	NIST	I	1	Number of intermediate mixing/homogenization steps between print locations.
	9	POUT1	R	1	Not used.
	10	ALFLIM	R	1	Limiting mixing rate.
	11	CMIXST	R	1	Mixing step size control parameter. $\alpha_{min} > \alpha_{lim} \Delta X = CMIXST*$
	12-17	DUMST	R	6	Not used.

Block Name (Link)	Words	Typical Variable Name	Type	Dim.	Description
SUPER (JETMIX)					Supersonic jet - Sonic line block
	1	SUPC	L	1	T = no supersonic points Reset to F after station processing.
	2	SUPSTP	L	1	T = no supersonic points.
	3	XSUP	L	1	X station where flow is fully subsonic.
TAG (JETMIX)					JETMLX Identification block
	1-10	NAME	I	10	User name.
	11-20	TITLE	R	10	Case title.
	21-30	IDENT	I	10	Case identification.
	31-40	ADDRES	R	10	User address.
	41-50	IDENT1	I	10	Extra case identification.
TCOF (PREJET, EMIS)					Polynomial coefficients for species thermo properties (NASA TN D-4097).
	1-350	C,TC	R	7,2, 25	Seven coefficients for high and low temperature ranges for each of up to 25 species.
	351	TLOW	R	1	High temperature range is from TMID to THI (°K).
	352	TMID	R	1	Low temperature range is from TLOW to TMID (°K).
	353	THI	R	L	
THERM (JETMIX)					Thermodynamic properties (vs Y).
	1-200	GMC	R	200	Heat capacity ratio $\gamma = C_p/C_v$.
	201-400	CP	R	200	Heat capacity at constant pressure, Btu/lbm °R.
THRST (JETMIX)					Integrated momentum (thrust) block.
	1-100	WV	R	100	Integrated momentum (thrust) at each station, ft lbm/sec ² .

Block Name (Link)	Words	Typical Variable Name	Type	Dim.	Description
TROUBL (MAIN)					PLUMOD error switches.
	1	ERR	L	1	T = program error (normally fatal).
	2	ERRMAJ	L	1	T = fatal program error.
	3	INERR	L	1	T = input error.
	4	PRERR	L	1	T = previous program error (not used).
UMESH (JETMIX)					Mesh or grid common block.
	1	MCHANG	L	1	T = do not redistribute mesh when the potential core disappears.
	2	CK	R	1	Mesh constant.
	3	DY1	R	1	ΔY (1st - 2nd mesh point) at point where potential core disappears.
	4	NMSH	I	1	Number of mesh points in the redistributed grid.
	5	CXPC	R	1	ΔX step size (potential core region).
	6	CXTP	R	1	ΔX step size (transition and similar region).
	7	NRED	I	1	Number of mesh points discarded when maximum points (NM) are reached.
XISAVE (EMIS)					Indexing info used by READT2.
	1	I	I	1	READT profile storage index.
	2	ISTAR	I	1	Number of last set of profile data read from JETMIX file.
XPRIN (JETMIX)					Profile print switch.
	1	DPRIN	L	1	T = print JETMIX profiles at every step (not recommended).
YOFXI (JETMIX, LSCK)					YOFX interpolation common
	1	IE	I	1	Saves interpolation index for use on the next entry.

Table 2. Key to Subscripts for Variables in
Common Blocks/GASCMP/and/GASTMW/.

Example: CONC (I, J, K, L)

I Species identification in order:

1	H	O	H ₂	O ₂	OH
6	H ₂ O	CO	CO ₂	N ₂	A
11	NO	N	HO ₂	NO ₂	NH ₃
16	C ₁₀ H _{10n}				

J Portion of 2-part heterogeneous gas (1 = hot,
 2 = cold)

K NEWNET tube number (Last is ambient air)

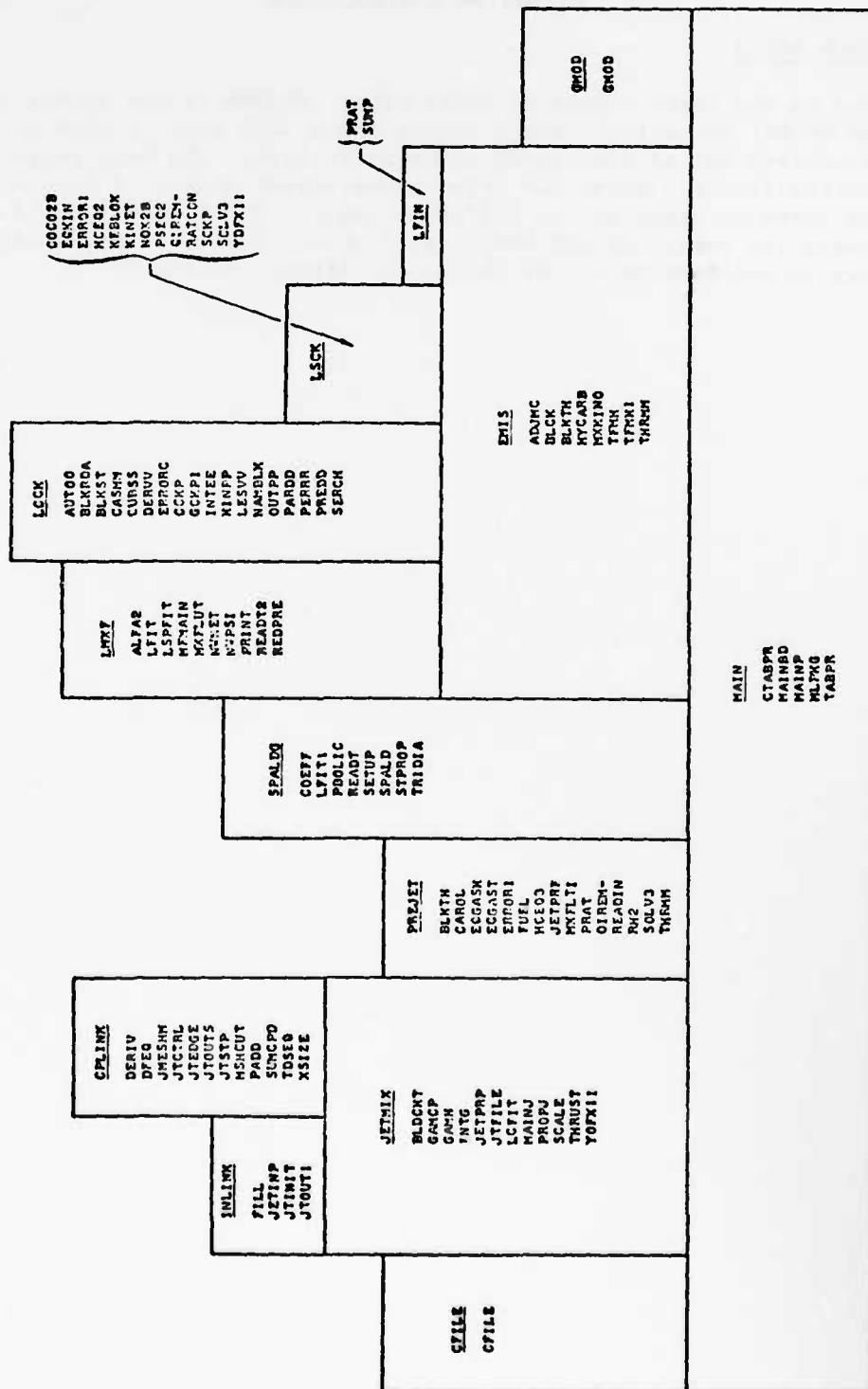
L Computation station (1 = previous, 2 = present)

2.3 Overlay Structure

To conserve computer memory, the program is run as an overlay job. The link structure is depicted in Table 3. Link names are underlined. The content of each link is described by deck labels, corresponding to the sub-program names in Sections 2.1 and 4.0. System library subroutines are not shown. The vertical heights of the boxes in Table 3 represent words of memory, roughly to scale. The horizontal axis corresponds to different times during execution of the program. The links are shown in the order that they exist on the loader input file (Section 3.1), not necessarily in the order in which they are executed.

The subprograms in each link are listed in alphabetical order in Table 3; however, it is important that the first decks in links JETMIX, PREJET, SPALDG, and EMIS be MAINJ, READIN, SPALD, and MISKINO, respectively.

Table 3. PLUMOD Overlay Diagram.



SECTION 3

OPERATING INSTRUCTIONS

3.1 Deck Setup

Due to the large number of subprograms, PLUMOD is not loaded from program decks, but rather from a loader input (R*) file created by the Object Library Editor and stored on magnetic tape. The file contains job control language cards and relocatable object decks. A list of the control cards is given on the following pages. The \$ OBJECT and \$ DKEND cards mark the beginning and end of each object deck. If necessary, the file may be modified by use of the Object Library Editor.

CONTENTS OF NEW OBJECT LIBRARY

3	USE	CD8AV/1/			
3	USE	CDHFC/31/			
3	USE	CHFL/888/			
3	OBJECT	IGNITION DELAY FOR RAM FUEL		F07.970080878MYCAR890	
		MYCAR8			
3	DKEND				
3	OBJECT	CALCULATE TEMPERATURE FROM ENTHALPY		F09.588080678TFMH0000	
		TFMH			
3	DKEND				
3	OBJECT	CP, T FROM M		F09.589080678TFMH0000	
		TFMH			
3	DKEND				
3	OBJECT	CALCULATES (DIMENSIONLESS) THERMODYNAMIC P		F12.945120171TMRM0000	
		TMRM			
3	DKEND				
3	OBJECT	BLKTH- BLOCK DATA (THERMO)		F10.318072578RLKTH000	
		ICUF			
3	DKEND				
3	OBJECT	BLCK- BLOCK DATA (SP/NN)		F15.713071178RLCK0000	
		NNNN			
3	DKEND				
3	OBJECT	MN OF RAM FUEL// ADJUST C2M4 THERMO COEF		F17.607110278ADJMC000	
		ADJMC			
3	DKEND				
3	LINK	LHXP			
3	OBJECT	-MFMAIN- MAIN SUB, FOR MXPLOT		F10.926062578MFMAIN00	
		MFMAIN			
3	DKEND				
3	OBJECT	READS JETTHIX- TAPE (MXPLOT)		F12.706063078HEADT200	
		READY			
3	DKEND				
3	OBJECT	REDISTRIBUTION OF STREAM FUNCTION FROM JET		F12.706063078HEADT200	
		NEPPI			
3	DKEND				
3	OBJECT	-REDPRE- SUB, ID READ, PREJET DATA		F08.597063078REDPRE00	
		REDPRE			
3	DKEND				
3	OBJECT	CONVERT JETTHIX PROFILES TO FIXED-NUMBER ST		F09.597063078REDPRE29	
		NEPNE7			
3	DKEND				
3	OBJECT	DUNNY--CALLS LEFT		F09.376111478LSPFI100	
		LSPFI7			
3	DKEND				
3	OBJECT	INTEGRATE OR INTERPOLATE		F11.248051178LFI10000	
		LFI7			
3	DKEND				
3	OBJECT	MXPLOT OUTPUT ROUTINE		F10.234062578PRINT000	
		MPRINT			
3	DKEND				
3	OBJECT			F10.234062578PRINT012	
				F17.658090578MXPLOT00	

	MXFLUT	SOLVE FOR MASS TRANSFER MATRIX	F09.917081974ALFZ2000	MXFLUT77 ALEA2000
3	DKEND	OBJECT	ALFA2	
3	DKEND	LOCK,LHXF		
3	LINK	DATA DRIVER ROUTINE FOR GCKP-1 CALC.	F19.382062774GCKP0000	GCKP0000
3	OBJECT	GCKP		
3	DKEND	GENERAL CHEMICAL KINETICS PROGRAM	F13.030062574GCKP1000	GCKP0012
3	OBJECT	GCKPI		
3	DKEND	STOICHIOMETRIC COEFFICIENTS	F16.793101973BLKST000	GCKPI015
3	OBJECT	SPEC		
3	DKEND	READ INPUT DATA AND INITIALIZE	F06.531050374KINPP000	BLKST025
3	OBJECT	KINP		
3	DKEND	-BLKRDA= BLOCK DATA (REACTION DAT	F15.050110874BLKRCA00	KINPP000
3	OBJECT	LBSRV		
3	DKEND	=NANBLK= BLOCK DATA (A/N)	F16.693010874NANBLK00	BLKRDA34
3	OBJECT	LUS		
3	DKEND	GENERAL OUTPUT, UNIT CONVERSIONS, ETC	F10.620032773OUTPP000	NANBLK05
3	OBJECT	OUTP		
3	DKEND	PERFORM ALL NECESSARY PRE-DERIVATIVE CALCU	F11.569032673PREDDC00	OUTPP211
3	OBJECT	PRED		
3	DKEND	COMPUTE ALL DERIVATIVES WRT THE INDEPENDEN	F11.576032673DERVV000	PRED0037
3	OBJECT	DERV		
3	DKEND	COMPUTE ALL MIXED PARTIAL DERIVATIVES	F20.568032673PARDC000	DERVV020
3	OBJECT	PARD		
3	DKEND	SET UP AND CALLFOR INTEGRATION	F13.053110073INTE000	PARD0075
3	OBJECT	INTE		
3	DKEND	CHOOSE STEP FORMULA, SET UP AUGMENTED MATR	F12.583103074CASWH000	INTE025
3	OBJECT	CASH		
3	DKEND	GENIL DBL PREC LINEAR EQN SOLVER	F12.957120171LLSV000	MCREMCASPMJ19
3	OBJECT	IESV		
3	DKEND	RELATIVE ERROR IN INTEGRATION STEP	F19.135110073ERRRC000	LESUV026
3	OBJECT	ERRRC		
3	DKEND	PREDICTS ERROR TO BE EXPECTED	F13.058110073PEAR000	ERRRC12
3	OBJECT	PEPR		

CONTENTS OF NEW OBJECT LIBRARY

3	DKEND	PERBR012
3	OBJECT	AUTOMATIC ELIMINATION FROM ERRDR CONSIDRA F12.966120171AUTOM000
3	DKEND	AUTOD
3	OBJECT	AUTOD0015
3	OBJECT	COEFFICIENTS FOR CURIC SOLINE INTERPOLATIO F12.970120171CURS000
3	DKEND	CURS
3	OBJECT	OPTIMAL SEQUENTIAL SEARCH TECHNIQUE F12.975120171SERCH000
3	DKEND	CJ83035
3	OBJECT	LINK LSCALOCK
3	OBJECT	HATN DRIVER FOR SCKP CALCULATION F19.347062774SCKP000
3	DKEND	SCKP
3	OBJECT	HYDROCARBON KINETICS ROUTINE F11.791112274K14ET000
3	DKEND	KINET
3	OBJECT	EQUILIBRIUM TERMS FOR KINETICS CALCULATI F11.809112274EQRIN000
3	DKEND	EQMIN
3	OBJECT	PSEQX F11.631112274K14LO00
3	DKEND	MC02
3	OBJECT	FULL-EQUILIBRIUM COMPOSITION CH(N)=AIR-M F17.595042175MCE0200
3	DKEND	PSE02
3	OBJECT	PSEUDO-EQUILIBRIUM COMPOSITION CH(N)=AIR-M F17.595042175MCE02051
3	DKEND	STE4
3	OBJECT	SOLVE 3 LINEAR EQUATIONS A(I,J)=B(I) F17.804031474SOLV000
3	DKEND	SOLV3
3	OBJECT	INTEGRATE NO RATE EQUATIONS F15.052110474NOX2800
3	DKEND	NOX28
3	OBJECT	COMPUTATION OF RATE CONSTANTS F19.353062774RATCON00
3	DKEND	RATCON
3	OBJECT	FINITE RATE CO/CO2 CHEMISTRY F08.602063075COC0200
3	DKEND	COC02
3	OBJECT	QUADRATIC INTERPOLATION ROOT EVALUATION F11.611070370IREM-00
3	DKEND	IREM
3	OBJECT	QIREM QUIT F12.962042375ERR0100
3	DKEND	ERR01
3	OBJECT	YOF F09.091080772YOFX100
3	DKEND	YOFX109

CONTENTS OF NEW OBJECT LIBRARY

3	EQUATE	ERROR1/CHEVR/	
3	LINK	LFIN,LSCK	
3	OBJECT	CALC. AND PRINT RESIDUAL EMISSIONS INDICES	F09.1220116758JMP0000
3		SUMUP	
3	DREND	OBJECT IMPACT/STATIC PRESSURE RATIO	PERFECT GAS F10.823100533PRAT0000
3		PRAT	
3	DREND		
3	LINK	CHDD,CHIS	
3	OBJECT	MODIFY INITIAL 6 PROFILES TO FORCE RAPID	F10.942062575GMD00000
3		CHDDCY	
3	DREND		10.942062575GMD00049
3	EXECUTE		
3	DISC	07	
3	DISC	40	
3	FILE	39	
3	TAPE	01	
3	DISC	02	
3	DISC	03	
3	DISC	04	

EOF ENCOUNTERED ON *R

The job control language required to run PLUMOD from the loader input tape is given below:

\triangle 1	\triangle 8	\triangle 16
\$	SNUMB	
\$	IDENT	
\$	USERID	
\$	EXECUTE	
\$	LIMITS	90, 41K, -4K
\$	TAPE	R*, R * D,, 35726 ,, R * PLUMOD
\$	SYSOUT	P*
\$	MASS	H *, H *R, 60R
\$	PRMFL	SL, R, R, AEG-LIB/SC-LB
\$	FILE	01, FIR, 10L
\$	FILE	02, F2R, 10L
\$	FILE	03, F3R, 10L
\$	TAPE	04, T4D,,, CASE-DESCR
\$	TAPDATA	04, \$R030
\$	FILE	39, F8R, 1L
\$	FILE	40, F9R, 1L
\$	INCODE	IBMF
(Input data)		
\$	ENDJOB	

The user must supply information on the \$ IDENT and \$ USERID cards. The R * tape number was current on date of publication, but is subject to change. Users are advised to check with the authors before proceeding. The permfile AEG-LIB/SC-LB is the AEG Scientific Library, from which MLPKG is drawn. Files 01, 02, 03, 39 and 40 are internal scratch files. File 04, assigned to output tape, will contain calculated data from tasks PREJET, JETMIX, GMOD, and SPALDG, and may be used for input to a subsequent PLUMOD run (as file 07) to re-start JETMIX or recompute SPALDG or EMIS with new parameters.

3.2 File Storage and Input Data Format

The PLUMOD system consists of six main calculation programs which are designated as follows:

<u>Program Name</u>	<u>Function</u>
PREJET	Problem input and initialization.
JETMIX	Free mixing calculation for the plume.
GMOD	Modification of initial "g" profiles.
SPALDG	Spalding "g" function calculation for the plume.
EMIS	Streamtube mixing. Chemical kinetics. Emission output.
CFILE	Merge or decollate output/input data stored on binary files.

The first five programs are normally run in a sequential fashion with output data passed to succeeding programs via binary output files. The program CFILE is used to merge the data produced by PREJET, JETMIX, and SPALDG for storage on a single file (usually magnetic tape) or to restore the merged data on the original files. The transfer files utilized in the PLUMOD system are:

File Code	Data Generated by:	Date Used by:
1	PREJET	JETMIX, GMOD, EMIS
2	JETMIX	SPALDG, EMIS
3	SPALDG	EMIS
4	CFILE	<u>Output</u> file with merged data from files 1, 2, 3
7	CFILE	<u>Input</u> file with merged data for restoration of files 1, 2, 3

As indicated above, the programs are normally run in a sequential fashion. The job may be terminated, however, after any program and the data saved on file 4 for a subsequent restart by running program CFILE. Also, CFILE may be initially run to restore data for either a restart (JETMIX only) or a continuation of the total PLUMOD job from the beginning of any program.

Card data are input to the PLUMOD system on file code I* (5) and consist of both formatted and Namelist data. The first three data cards are used for case identification and are read with a 1X,10A6 format by MAINP; viz,

2

(Name - up to 60 alphanumeric characters)

(Address - up to 60 alphanumeric characters)

(Ident. - up to 60 alphanumeric characters)

The problem input to the system follows these cards and consists of a program card designating the program name and the existence of an input file and/or an output file. The general format of the program card is as follows:

	(input file ?)	(output file?)
2	12	14
(Program name)	(T or F)	(T or F)

In all cases, the program card is followed by one (1) or more Namelist sets which are read by the calculation programs. A typical input card deck to run the entire PLUMOD system and save an output file is shown below:

2	12	14
PREJET	F	T
\$TSTDAT	(Namelist data for READIN) \$	
\$A	(Namelist data for JETPRF) \$	
JETMIX	T	T
\$A	(Namelist data for JETINP) \$	
GMOD	T	T
\$A	(Namelist data for GMOD) \$	
SPALDG	T	T
\$CHANGE	(Namelist data for SPALD) \$	
EMIS	T	F
\$A	(Namelist data for MXKINO) \$	
\$INPUT	(Namelist data for MFMAIN) \$	
\$PROB	(Namelist data for KINPP, if GCKP is used) \$	
CFILE	T	T
\$A	(Namelist data for CFILE) \$	

Note that the card input to CFILE may follow any of the individual input sets if it desired to only run a portion of the problem and save the cumulative output on file 4. Also, if the problem is to be restarted (JETMIX) or continued, the CFILE input must precede the card input to the program(s)

to be executed. Since the PLUMOD system normally requires a large amount of processing time and utilizes a large amount of computer resources, no provision exists for the execution of multiple cases.

Detailed descriptions of the variables in the various Namelists are given in the table below. Here, BITS means the Honeywell standard noise word (0377 777 777 777). The "initial value" is the value assigned to the variable by the program prior to reading the input. The "default value" is the value given the variable after reading input in the event the initial value is not changed. In cases where internal initialization is not provided, recommended values are given where appropriate. For rules concerning input of Namelist data, the reader is referred to the Honeywell Fortran IV manual (Reference 8)

NAMELIST / TSTDAT / Engine test data read by READIN

<u>Variable Name</u>	<u>Type</u>	<u>Dim.</u>	<u>Init. Value</u>	<u>Default Value</u>	<u>Description</u>
PO	R	1	14.696	-	Ambient air static pressure, psia.
TO	R	1	518.69	-	Ambient air static temperature, ° R
HUM	R	1	.00531	-	Air specific humidity, lb moisture/lb dry air
VO	R	1	0.0	-	Flight speed, ft/sec
RADJ	R	1	1.0	-	Outer radius of exhaust jet, ft
AMBT	R	5			Equivalent to PO thru RADJ
T2	R	1	518.69	-	Ram air total temperature, ° R
AR5	R	1	0.02	-	Turbine exit fuel-air ratio, lb fuel/lb dry air
EINO2C	R	1	20.0	-	Main combustor NOX emission index, lb NO ₂ /Klb fuel
BETA	R	1	0.0	-	Engine bypass ratio, lb fan air/lb core air
T25	R	1	518.69	-	Fan discharge temperature, ° R
CYCLE	R	5			Equivalent to T2 thru T25
FUEL	R	3	2.0	-	Hydrogen-carbon atom ratio in fuel
			537.0	-	Fuel temperature, ° R
			BITS	Estimated	Lower heating value of fuel, B

NAMELIST / TSTDAT / (Continued)

<u>Variable Name</u>	<u>Type</u>	<u>Dim.</u>	<u>Init. Value</u>	<u>Default Value</u>	<u>Description</u>
CAROL	R	4, 11	BITS	-	CAROL gas analyses of CO, CO ₂ , Hydrocarbon, and NO _x
RADII	R	11	BITS	-	Radial locations of probe measurements
PT	R	11	BITS	-	Probe impact pressure measurements, psia
PS	R	11	BITS	PO	Static pressure at probe locations, psia
BLOC	R	11	BITS	BETA	Local bypass ratio at probe locations
EICOC	R	11	BITS	0.0	Main combustor CO Emission Index, lb CO/klb fuel
ALDAT	R	10, 11	BITS		Alternate vector for CAROL thru EICOC info. If both are given, CAROL etc replace values in ALDAT. Example: PT (n), if given, replaces ALDAT (6, n). ALDAT (10, n) not used.
TITLE	H	20	Blank	-	Output page heading information
SF	R	1	1.0	-	Scale factor for CAROL data (leave 1.0 if analysis given in mole fractions, set 1E-6 if in parts per million, etc)
PRINT	R	30	BITS	0.1*RADJ 10*RADJ BITS	Axial stations (feet) at which output is desired

NAMelist / A / JETMIX parameters and controls read by JETPRF

<u>Variable Name</u>	<u>Type</u>	<u>Dim.</u>	<u>Init. Value</u>	<u>Recomm. Value</u>	<u>Description</u>		
TIJET	R	1	-	0.02	Initial jet turbu- lence intensity (u_p'/u_p)		
TIE	R	1	-	0.0	External (ambient) turbulence intensity		
RG	R	1	-	53.34	Gas constant (ft lbf/lbm ° R)		
PR	R	1	-	0.72	Laminer Prandtl number		
PRT	R	1	-	0.70	Turbulent Prandtl number		
SC	R	1	-	201.6	Sutherland constant, ° R		
TREF	R	1	-	0.0	Reference tempera- ture for viscosity, ° R		
MUREF	R	1	-	0.0	Reference viscosity, lbm/ft sec		
NM	I	1	-	95	Maximum number of JETMIX streamlines before mesh redistri- bution occurs		
CA	R	1	-	6.4303	For jet dummy gases	}	*
CB	R	1	-	8.929E-4			
CC	R	1	-	5.989E-8			
CAA	R	1	-	6.4303	For ambient air	}	
CBA	R	1	-	8.929E-4			
CCA	R	1	-	5.989E-8			

*Coefficients for determination of specific heat as a quadratic function of local static temperature (° R). $cp = a + bT + cT^2$

NAMelist / A / (JETPRF - Continued)

<u>Variable Name</u>	<u>Type</u>	<u>Dim.</u>	<u>Init. Value</u>	<u>Recomm. Value</u>	<u>Description</u>
CT1	R	1	-	0.23	} Constants in the empirical equation for turbulence length scale in the potential core region
CT2	R	1	-	0.38	
CT3	R	1	-	0.23	} Constants in the empirical equation for turbulence length scale in the transition region
CT4	R	1	-	0.05	
CT5	R	1	-	0.38	
CT6	R	1	-	1.4	
CT7	R	1	-	0.43	
CT8	R	1	-	0.1875	Constant in the empirical equation for turbulence length scale in the fully developed region

NAMELIST / A / JETMIX parameters and controls read by JETINP

<u>Variable Name</u>	<u>Type</u>	<u>Dim.</u>	<u>Init. Value</u>	<u>Recomm. Value</u>	<u>Description</u>
DIAJ	R	1	-	*	Diameter (AXI=T) or 2* half height of primary jet, in
MJET	R	1	BJTS	*	Primary jet Mach number
TJET	R	1	518.688	*	Primary jet static temperature, ° R
PTJET	R	1	BITS	*	Primary jet stagnation pressure, psia
VJET	R	1	-	*	Primary jet velocity, fps (u_j)
TIJET	R	1	0.	*	Primary jet turbulence intensity (Referenced to YJET)
PE	R	1	14.69594	*	Ambient pressure, psia
VE	R	1	-	*	External stream velocity, fps
ME	R	1	-	*	External stream Mach number
TIE	R	1	0.	*	External stream turbu- lence intensity (Referenced to VJET)
TE	R	1	518.688	*	Ambient temperature, ° R
AXI	L	1	T	-	T = axisymmetric, F= plane (2 - D)
NJ	I	1	-	*	Number of mesh points in primary jet initial profile
NM	I	1	95	*	Maxumum number of mesh points
GAM	R	1	-	-	Not used

NAMELIST / A / (JETINP - Continued)

<u>Variable Name</u>	<u>Type</u>	<u>Dim.</u>	<u>Init. Value</u>	<u>Recomm. Value</u>	<u>Description</u>
RG	R	1	53.34 (air)	*	Gas constant, ft lb _f /lbm ° R
PR	R	1	0.72 (air)	*	Prandtl number
PRT	R	1	1.0	*	Turbulent Prandtl number
X	R	100	-	**	Dimensionless stream-wise coordinate (free mixing) $X = x/d_j$
XPRN	I	100	0	2.0	Profile print indicator ≥ 1 = Print, 0 = Do not print
SC	R	1	BITS	*	Sutherland constant for viscosity calculation, ° R. If not input, air values are used.
TREF	R	1	BITS	*	Reference temperature for viscosity calculation, ° R
MUREF	R	1	BITS	*	Reference viscosity, lbm/ft sec
SP	R	1	-	-	Not used
SV	R	1	-	-	Not used
SLEN	R	1	-	-	Not used
DPRIN	L	1	F	-	T - print profiles at each step
PLOT	L	1	-	-	Not used
NRED	I	1	-	-	Not used
PM	R	10	-	-	Not used
CXPC	R	1	.02	0.15**	Step size control - potential core region. Fraction % of mixing zone width

NAMELIST / A / (JETINP - Continued)

<u>Variable Name</u>	<u>Type</u>	<u>Dim.</u>	<u>Init. Value</u>	<u>Recomm. Value</u>	<u>Description</u>
CXTP	R	1	.02	.015	Step size control - transition/similar region. Fractional % of jet radius or half height.
MCHANG	L	1	-	-	Not used
IDENT	H	10	Blank	*	Identification block
Y	R	200	-	*	Dimensionless normal coordinate $Y = y/r_J$
UD	R	200		*	Dimensionless velocity $UD = u/u_J$
THD	R	200	-	*	Dimensionless temper- ature, $THD = T/T_J$
ED	R	200	-	*	Dimensionless turbulence energy, $ED = e/e_J$
TID	R	200	-	*	Turbulence intensity (Reference to VJET)
CT1	R	1	-	*	<div style="display: flex; align-items: center;"> <div style="flex: 1;">Turbulence intensity</div> <div style="flex: 1; text-align: center;"> <div style="border-left: 1px solid black; height: 100px; margin: 0 auto;"></div> <div>(See input to JETPRF)</div> <div style="border-left: 1px solid black; height: 100px; margin: 0 auto;"></div> <div style="margin-top: 5px;">↓</div> </div> </div>
CT2	R	1	-	*	
CT3	R	1	-	*	
CT4	R	1	-	*	
CT5	R	1	-	*	
CT6	R	1	-	*	
CT7	R	1	-	*	
CT8	R	1	-	*	
CT9	R	1	-	-	
CTM	R	1	.23	-	Coannular mixing, turbulence scale pro- portionality constant for merged region

NAMELIST / A / (JETINP - Continued)

<u>Variable Name</u>	<u>Type</u>	<u>Dim.</u>	<u>Init. Value</u>	<u>Recomm. Value</u>	<u>Description</u>
CTS	R	1	.23	-	Coannular mixing, turbulence scale proportionality constant for secondary jet
CTP	R	1	.175	-	Coannular mixing, turbulence scale proportionality constant for primary jet
B	R	500	-	-	Not used
NB	I	1	-	-	Not used
TAB	R	5	-	-	Not used
D	R	800	-	-	Not used
ND	I	1	-	-	Not used
TAD	R	4	-	-	Not used
RESTR	R	1	BITS	**	X station for restart of problem
CK	R	1	1.06064475	1.0466475	Mesh parameters for transition, similar region $\psi_i = \frac{DY1((CK)^{i-1}-1)}{(CK-1)} \quad i = 1, NMSH$
DY1	R	1	.001	.002	
NMSH	I	1	71	-	
MIX	L	1	F	-	Not used
MAXIT	I	1	-	-	Not used
CF	R	1	-	-	Not used
TOL	R	1	-	-	Not used
SUPB	L	1	-	-	Not used
RD	R	100	-	-	Not used
XD	R	100	-	-	Not used
YCB	R	100	-	-	Not used

NAMELIST / A / (JETINP - Continued)

<u>Variable Name</u>	<u>Type</u>	<u>Dim.</u>	<u>Init. Value</u>	<u>Recomm. Value</u>	<u>Description</u>
TWO	L	1	F	-	Jet configuration - F - single mixing region T - coannular mixing region
DIAO	R	1	-	-	Diameter (AXI = T) or 2* half height of secondary jet, in.
MJETO	R	1	BITS	-	Secondary jet Mach number
TJETO	R	1	518.688	-	Secondary jet static temperature, ° R
VJETO	R	1	-	-	Secondary jet velocity, fps
PTJETO	R	1	BITS	-	Secondary jet stagnation pressure, psia
TIJETO	R	1	0.	-	Secondary jet turbulence intensity (Relative to VJET)
NJO	I	1	-	-	Mesh index of outermost point in secondary jet
NC	I	1	12	*	Number of active constituents
CNAME	H	12	12	*	Names of active constituents
ALJ	R	12	-	*	Primary jet stream species mole fractions
ALJO	R	12	-	-	Secondary jet stream species mole fractions
ALE	R	12	-	*	External (boundary) species mole fractions
SCM	R	12	.7	*	Effective Schmidt number for individual species

NAMELIST / A / (JETINP - Continued)

<u>Variable Name</u>	<u>Type</u>	<u>Dim.</u>	<u>Init. Value</u>	<u>Recomm. Value</u>	<u>Description</u>
TCPRF	R	-	-	-	Not used
HCPRF	R	-	-	-	Not used
CPC	R	(3, 12)	-	*	Coefficients in polynomial representation of species molar heat capacities
ALX	R	(100, 12)	-	*	Species mole fraction profiles
DIFF	L	1	T	-	Species diffusion - F - no, T - yes
CSC	R	1	1000.	-	Scaling constant for file storage
NDIGIT	I	1	-	-	Not used
LTERP	L	1	T	-	Transition region scale interpolation T - linear F - exponential
CDIFF	-	1	-	-	Not used
CHI	R	1	.586	*	Diffusion parameter (turbulence energy equation)
SCLD	L	1	F	T	T - Use ambient gas mole fraction to determine inner edge of mixing zone F - Use velocity to determine inner edge of mixing zone
ALXLIM	R	1	.0001	-	Limiting ambient gas mole fraction used to locate inner edge of mixing zone

NAMELIST / A / (JETINP - Continued)

<u>Variable Name</u>	<u>Type</u>	<u>Dim.</u>	<u>Init. Value</u>	<u>Recomm. Value</u>	<u>Description</u>
CJRMOD	R	1	-	1.0	Fraction of jet radius used as reference length in determining mixing scale within potential core

* Transmitted from JETPRF

** See Section 3.4

NAMELIST / A / Specifications read by GMOD

<u>Variable Name</u>	<u>Type</u>	<u>Dim.</u>	<u>Init. Value</u>	<u>Default Value</u>	<u>Description</u>
PDUM	R	20	-	-	If PDUM (1) \neq 0., modified profile parameters are printed
YAIR	R	1	BITS	0.0	Mass fraction of ambient air regarded as "hot" gas
YMIN	R	1	0.30	-	Minimum mass fraction hot gas at any point

NAMelist / CHANGE / Parameters and controls read by SPALD

<u>Variable Name</u>	<u>Type</u>	<u>Dim.</u>	<u>Init. Value</u>	<u>Recomm. Value</u>	<u>Description</u>
ALFA	R	1	-	0.2	Coefficients in Spald- ing heterogeneity con- servation equation (Reference 1, Section 4.4)
CG1	R	1	-	2.7	
CG2	R	1	-	0.67	
SIGMAC	R	1	-	0.7	
THETA	R	1	-	0.5	Weighting factor for finite-difference solution
DX	R	1	1000.	0.01	Initial axial step length, jet diameters
DXPRN	R	1	-	1.0	Not used
XSTOP	R	1	1000.	24.0	Last axial station (x/d_j) at which heterogeneity param- eter will be computed
NSZ1	I	1	-	25	Number of equally spaced streamlines at initial axial station
PRINT	L	1	F	-	Diagnostic profiles of G are printed

NAMelist / A / Control variables read by MXKINO

<u>Variable Name</u>	<u>Type</u>	<u>Dim.</u>	<u>Init. Value</u>	<u>Recomm. Value</u>	<u>Description</u>
PDUM	R	20	0.0	-	Diagnostic printout controls for KINET, EQKIN, MFMAIN, READT, MXFLUT, SCKP, HYCAB, GCKP
CHEMK	I	1	1	2	1 means use GCKP; 2 means use SCKP
PSTA	R	30	BITS	-	Axial stations (jet diameters) at which output is desired. Overrides values in \$TSTDAT.
NIST	I	1	2	-	Not used
COREQ	L	1	-	T	Use current rate equation for CO con- sumption reaction in SCKP

NAMELIST / INPUT / Control variables read by MFMAIN

<u>Variable Name</u>	<u>Type</u>	<u>Dim.</u>	<u>Init. Value</u>	<u>Recomm. Value</u>	<u>Description</u>
X	R	1	0.0	-	Do not use
XL	R	1	1.2	-	Do not use
NTUBES	I	1	-	-	Do not use
DX	R	1	.05	-	Mixing step size, jet diameters
ALFLIM	R	1	-	-	Inoperable
CMIXST	R	1	2.0	-	Factor for automatic increase in DX
NIST	I	1	10	-	Number of mixing steps per reaction step

NAMELIST / PROB / GCKP Control variables used by KINPP (Include only if
MXKINO \$A CHEMK = 1)

<u>Variable Name</u>	<u>Type</u>	<u>Dim.</u>	<u>Init. Value</u>	<u>Recomm. Value</u>	<u>Description</u>
HMIN	R	1	0.5E-7	1.3E-8	Minimum integration step size, sec.
HMAX	R	1	0.5E-4	1E-4	Maximum step size, sec.
HINT	R	1	0.0	2.6E-8	Initial step size, sec.
EMAX	R	1	0.0001	0.01	Maximum projected relative error, used for internal step size adjustment
ALLM1	L	1	F	F	All third-body collision effective- nesses are 1.0
ELIM	L	1	F	-	Not used
CONC	L	1	T	-	" "
EXCHR	L	1	F	-	" "
IPRCOD	I	1	3	-	" "
ITPSZ	I	1	5	-	" "
XTB	R	40	0.0	-	" "
ATB	R	40	0.0	-	" "
NTB	I	1	0	-	" "
CX3	R	1	0.0	-	" "
CX2	R	1	0.0	-	" "
CX1	R	1	0.0	-	" "
CX0	R	1	0.0	-	" "
LSUBM	R	1	0.0	-	" "

NAMELIST / PROB / (Continued)

<u>Variable Name</u>	<u>Type</u>	<u>Dim.</u>	<u>Init. Value</u>	<u>Recomm. Value</u>	<u>Description</u>
ETA					Not used
D	R	1	0.0	-	" "
VISC	R	1	0.0	-	" "
BETA	R	1	0.0	-	" "
END	R	1	0.0	-	" "
DELP	R	1	0.0	-	" "
PRINT	R	50	-	-	" "
NPRNTS	I	1	0	-	" "
APRINT	R	50	0.0	-	" "
EVSTEP	L	1	F	-	" "
DEBUG	L	1	F	-	" "
COMBUS	L	1	F	-	" "
SHOCK	L	1	F	-	" "
TCON	L	1	F	-	" "
RHOCON	L	1	F	-	" "
BRIEF	L	1	F	-	" "
TIMLMT	R	1	25.0	-	" "

NAMELIST / A / File manipulation input read by CFILE

<u>Variable Name</u>	<u>Type</u>	<u>Dim.</u>	<u>Init. Value</u>	<u>Default Value</u>	<u>Description</u>
MERGE	L	1	F	-	T = Merge files (store on file code 4)
SPLIT	L	1	F	-	T = Restore files (restore files stored on file code 7)
NF	I	1	0	-	File(s) present indicator: =1 PREJET data =2 PREJET & JETMIX data =3 PREJET, JETMIX, SPALDG data

3.3 Sample Calculation

The following input data cards, representing data from a test of a J85-5 engine at mid AB power setting (Reference 1), were input to the plume model:

137

Using the input data given above, PLUMOD generated the computation reports shown below. For brevity, JETMIX reports at the first and last computation stations only are reproduced here. The SPALDG reports at only the initial stations are shown, and some EMIS reports have been omitted.

* FREE JET PROGRAM *
 * ISOTHERMAL * * * * * COMPRESSIBLE * JET

** SINGLE MIXING REGION **
 WHEN DATE PERCUSION
 ADDRESS EVENDALE OHIO

* INPUT AND INITIAL CONDITIONS *
 EXTERNAL CONDITIONS JET DISCHARGE PARAMETERS GAS PROPERTIES
 TE = 515.000 DIAJ = 16.20000 GAM = 1.40000
 PE = 1.3850E-01 VJET = 1.0089 PC = 53.33600
 VE = 0. VJET = 2330.982 PR = 0.72000
 ME = 0. PTJET = 2.6014E 01 PRT = 0.70000
 TIE = 0. TJET = 2503.983 SC =
 TJET = 2.0000E-02 TREF =
 FLUMJ = 4.4609E 00 MUREFO

N	V	PGT	U	TMETA	T1	GAS1	GAS2	GAS3	GAS4	GAS5	GAS6
1	0.	0.	0.92308828	0.88319692	2.0000000E-02	1.0000E 00	0.	0.	0.	0.	0.
2	1.0000E-02	0.1287E-01	0.92308828	0.88319692	2.0000000E-02	1.0000E 00	0.	0.	0.	0.	0.
3	7.5524E-02	2.3265E-05	0.92308828	0.88319692	2.0000000E-02	1.0000E 00	0.	0.	0.	0.	0.
4	1.1117E-01	8.2203E-05	0.92308828	0.88319692	2.0000000E-02	1.0000E 00	0.	0.	0.	0.	0.
5	1.4617E-01	6.9134E-05	0.93226840	0.89063979	2.0000000E-02	9.0000E-01	1.0000E-01	0.	0.	0.	0.
6	1.8117E-01	9.8293E-05	0.96897890	0.92041126	2.0000000E-02	5.0000E-01	5.0000E-01	0.	0.	0.	0.
7	1.4617E-01	1.0069E-04	1.00569138	0.95018272	2.0000000E-02	1.0000E-01	1.0000E-01	0.	0.	0.	0.
8	1.8117E-01	1.0733E-04	1.01086951	0.95762558	2.0000000E-02	0.	1.0000E 00	0.	0.	0.	0.
9	2.2972E-01	2.1940E-04	1.01886951	0.95762558	2.0000000E-02	0.	1.0000E 00	0.	0.	0.	0.
10	2.9828E-01	3.7078E-04	1.01886951	0.95762558	2.0000000E-02	0.	1.0000E 00	0.	0.	0.	0.
11	3.0328E-01	3.7078E-04	1.02151354	0.96433187	2.0000000E-02	0.	1.0000E-01	1.0000E-01	0.	0.	0.
12	3.0428E-01	3.7613E-04	1.04808968	0.99123671	2.0000000E-02	0.	5.0000E-01	5.0000E-01	0.	0.	0.
13	3.1324E-01	4.0409E-04	1.07465582	1.01916159	2.0000000E-02	0.	1.0000E-01	9.0000E-01	0.	0.	0.
14	3.1924E-01	4.2224E-04	1.08130984	1.02496783	2.0000000E-02	0.	0.	1.0000E 00	0.	0.	0.
15	3.2524E-01	4.5093E-04	1.08130984	1.02496783	2.0000000E-02	0.	0.	1.0000E 00	0.	0.	0.
16	4.7213E-01	9.2697E-04	1.08130984	1.02496783	2.0000000E-02	0.	0.	1.0000E 00	0.	0.	0.
17	4.7213E-01	9.2697E-04	1.08130984	1.02496783	2.0000000E-02	0.	0.	1.0000E 00	0.	0.	0.
18	4.8221E-01	9.8857E-04	1.12064899	1.07278134	2.0000000E-02	0.	0.	1.0000E-01	1.0000E-01	0.	0.
19	4.8723E-01	9.8848E-04	1.12064899	1.07278134	2.0000000E-02	0.	0.	5.0000E-01	5.0000E-01	0.	0.
20	4.9223E-01	1.0083E-03	1.15210952	1.11109618	2.0000000E-02	0.	0.	1.0000E-01	9.0000E-01	0.	0.
21	5.0930E-01	1.3455E-03	1.15997816	1.12067485	2.0000000E-02	0.	0.	0.	1.0000E 00	0.	0.
22	6.0754E-01	1.7313E-03	1.15997816	1.12067485	2.0000000E-02	0.	0.	0.	1.0000E 00	0.	0.
23	6.5254E-01	1.7541E-03	1.15997816	1.12067485	2.0000000E-02	0.	0.	0.	1.0000E 00	0.	0.
24	6.5254E-01	1.7541E-03	1.15997816	1.12067485	2.0000000E-02	0.	0.	0.	1.0000E 00	0.	0.
25	6.5254E-01	1.7541E-03	1.15997816	1.12067485	2.0000000E-02	0.	0.	0.	1.0000E 00	0.	0.
26	6.5254E-01	1.7541E-03	1.15997816	1.12067485	2.0000000E-02	0.	0.	0.	1.0000E 00	0.	0.
27	7.1047E-01	2.0514E-03	1.14499070	1.09630102	2.0000000E-02	0.	0.	0.	1.0000E 00	0.	0.
28	7.5337E-01	2.3622E-03	1.14499070	1.09630102	2.0000000E-02	0.	0.	0.	1.0000E 00	0.	0.
29	7.5437E-01	2.3733E-03	1.13501762	1.08825634	2.0000000E-02	0.	0.	0.	9.0000E-01	1.0000E-01	0.

INITIAL PROFILES

N	Y	Q51	GAS7	GAS8	GAS9	GAS10	GAS11	GAS12
1	0.	0.	0.	0.	0.	0.	0.	0.
2	1.3000E-02	0.1207E-07	0.	0.	0.	0.	0.	0.
3	1.5500E-02	2.365E-05	0.	0.	0.	0.	0.	0.
4	1.6117E-01	6.2803E-05	0.	0.	0.	0.	0.	0.
5	1.6617E-01	6.6133E-05	0.	0.	0.	0.	0.	0.
6	1.7117E-01	6.8693E-05	0.	0.	0.	0.	0.	0.
7	1.7617E-01	1.0569E-04	0.	0.	0.	0.	0.	0.
8	1.8117E-01	1.2791E-04	0.	0.	0.	0.	0.	0.
9	1.8617E-01	1.5043E-04	0.	0.	0.	0.	0.	0.
10	1.9117E-01	1.7317E-04	0.	0.	0.	0.	0.	0.
11	1.9617E-01	1.9613E-04	0.	0.	0.	0.	0.	0.
12	2.0117E-01	2.1923E-04	0.	0.	0.	0.	0.	0.
13	2.0617E-01	2.4243E-04	0.	0.	0.	0.	0.	0.
14	2.1117E-01	2.6573E-04	0.	0.	0.	0.	0.	0.
15	2.1617E-01	2.8913E-04	0.	0.	0.	0.	0.	0.
16	2.2117E-01	3.1263E-04	0.	0.	0.	0.	0.	0.
17	2.2617E-01	3.3623E-04	0.	0.	0.	0.	0.	0.
18	2.3117E-01	3.5993E-04	0.	0.	0.	0.	0.	0.
19	2.3617E-01	3.8373E-04	0.	0.	0.	0.	0.	0.
20	2.4117E-01	4.0763E-04	0.	0.	0.	0.	0.	0.
21	2.4617E-01	4.3163E-04	0.	0.	0.	0.	0.	0.
22	2.5117E-01	4.5573E-04	0.	0.	0.	0.	0.	0.
23	2.5617E-01	4.7993E-04	0.	0.	0.	0.	0.	0.
24	2.6117E-01	5.0423E-04	0.	0.	0.	0.	0.	0.
25	2.6617E-01	5.2863E-04	0.	0.	0.	0.	0.	0.
26	2.7117E-01	5.5313E-04	0.	0.	0.	0.	0.	0.
27	2.7617E-01	5.7773E-04	0.	0.	0.	0.	0.	0.
28	2.8117E-01	6.0243E-04	0.	0.	0.	0.	0.	0.
29	2.8617E-01	6.2723E-04	0.	0.	0.	0.	0.	0.
30	2.9117E-01	6.5213E-04	0.	0.	0.	0.	0.	0.
31	2.9617E-01	6.7713E-04	0.	0.	0.	0.	0.	0.
32	3.0117E-01	7.0223E-04	0.	0.	0.	0.	0.	0.
33	3.0617E-01	7.2743E-04	0.	0.	0.	0.	0.	0.
34	3.1117E-01	7.5273E-04	0.	0.	0.	0.	0.	0.
35	3.1617E-01	7.7813E-04	0.	0.	0.	0.	0.	0.
36	3.2117E-01	8.0363E-04	0.	0.	0.	0.	0.	0.
37	3.2617E-01	8.2923E-04	0.	0.	0.	0.	0.	0.
38	3.3117E-01	8.5493E-04	0.	0.	0.	0.	0.	0.
39	3.3617E-01	8.8073E-04	0.	0.	0.	0.	0.	0.
40	3.4117E-01	9.0663E-04	0.	0.	0.	0.	0.	0.
41	3.4617E-01	9.3263E-04	0.	0.	0.	0.	0.	0.
42	3.5117E-01	9.5873E-04	0.	0.	0.	0.	0.	0.
43	3.5617E-01	9.8493E-04	0.	0.	0.	0.	0.	0.
44	3.6117E-01	1.0112E-03	0.	0.	0.	0.	0.	0.
45	3.6617E-01	1.0382E-03	0.	0.	0.	0.	0.	0.
46	3.7117E-01	1.0652E-03	0.	0.	0.	0.	0.	0.
47	3.7617E-01	1.0922E-03	0.	0.	0.	0.	0.	0.
48	3.8117E-01	1.1192E-03	0.	0.	0.	0.	0.	0.
49	3.8617E-01	1.1462E-03	0.	0.	0.	0.	0.	0.
50	3.9117E-01	1.1732E-03	0.	0.	0.	0.	0.	0.
51	3.9617E-01	1.2002E-03	0.	0.	0.	0.	0.	0.
52	4.0117E-01	1.2272E-03	0.	0.	0.	0.	0.	0.
53	4.0617E-01	1.2542E-03	0.	0.	0.	0.	0.	0.
54	4.1117E-01	1.2812E-03	0.	0.	0.	0.	0.	0.
55	4.1617E-01	1.3082E-03	0.	0.	0.	0.	0.	0.
56	4.2117E-01	1.3352E-03	0.	0.	0.	0.	0.	0.
57	4.2617E-01	1.3622E-03	0.	0.	0.	0.	0.	0.
58	4.3117E-01	1.3892E-03	0.	0.	0.	0.	0.	0.
59	4.3617E-01	1.4162E-03	0.	0.	0.	0.	0.	0.
60	4.4117E-01	1.4432E-03	0.	0.	0.	0.	0.	0.

60	9.000E-01	3.653E-03	0.	0.	0.	5.000E-01	5.000E-01	0.
61	9.000E-01	3.653E-03	0.	0.	0.	1.000E-01	9.000E-01	0.
62	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
63	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
64	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
65	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
66	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
67	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
68	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
69	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
70	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
71	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
72	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
73	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
74	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
75	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
76	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
77	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
78	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
79	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
80	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
81	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
82	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
83	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
84	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
85	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
86	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
87	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
88	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
89	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
90	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
91	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
92	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
93	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
94	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
95	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
96	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
97	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
98	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
99	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.
100	9.000E-01	3.653E-03	0.	0.	0.	0.	1.000E-01	0.

007415 IS THE LAST TIME THE ABOVE MESSAGE WILL APPEAR

JET ANALYSIS PROGRAM

PROFILES-- STA (331) X= 25.00000 PRESSURE= 13.0300

** DIMENSIONLESS **														** DIMENSIONAL **	
Y														7	
PSI														U	
UD														MACH	
7WD														PTD	
TI														TTD	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	
0.375737														0.30776	
0.10205-02														0.160278	
0.17637														0.034503	

PROFILES-- STA (33) X= 25.0000 PRESSURE# 13.6300

Y	Y	P91	UD	IMO	DI	DI	DI	PTD	MACH	U	I	DI	P707
** DIMENSIONLESS **													
51	3.30755	1.7047E-02	0.101939	0.326704	3.72214E-02	0.1136999	0.0233315	0.13870	255.2541	761.5311	769.1524	13.9726	
52	3.35460	1.7013E-02	0.098598	0.324779	3.66733E-02	0.1109864	0.0224499	0.13820	246.8666	755.8889	763.0674	13.9529	
53	3.40021	1.6981E-02	0.095195	0.322795	3.60281E-02	0.1081376	0.0210580	0.13738	238.2407	743.2924	756.7417	13.9327	
54	3.44627	1.6951E-02	0.091577	0.319096	3.53395E-02	0.1052069	0.0196610	0.13653	229.3076	731.6074	750.1096	13.9129	
55	3.49402	1.6924E-02	0.087891	0.315324	3.47404E-02	0.1021271	0.0182594	0.13535	220.0781	720.3054	743.2557	13.8926	
56	3.53197	1.6903E-02	0.084083	0.311420	3.39850E-02	0.0989096	0.0168573	0.13491	210.5420	709.5172	736.2744	13.8723	
57	3.56592	1.6882E-02	0.080198	0.307376	3.31648E-02	0.0955452	0.0154845	0.13422	200.6891	700.4805	728.5543	13.8524	
58	3.59521	1.6861E-02	0.076080	0.303179	3.22799E-02	0.0920227	0.0140670	0.13324	190.5033	692.0247	720.6513	13.8323	
59	3.62046	1.6840E-02	0.071874	0.300077	3.13198E-02	0.0883293	0.0126875	0.13197	180.2752	684.1947	712.6261	13.8125	
60	3.64182	1.6819E-02	0.067527	0.296537	3.02740E-02	0.0844496	0.0113249	0.13039	169.0854	676.4399	703.7546	13.7929	
61	3.65924	1.6801E-02	0.063025	0.292475	2.91494E-02	0.0803650	0.0099948	0.12846	157.8145	668.2221	694.4250	13.7734	
62	3.67276	1.6781E-02	0.058362	0.287975	2.79212E-02	0.0760521	0.0086736	0.12616	146.1377	660.1975	684.9853	13.7544	
63	3.68203	1.6761E-02	0.053524	0.283061	2.66419E-02	0.0714814	0.0073982	0.12346	134.0234	652.1081	674.7591	13.7354	
64	3.68821	1.6740E-02	0.048494	0.277918	2.51131E-02	0.0666136	0.0061666	0.12043	121.4291	643.6744	663.8490	13.7167	
65	3.69022	1.6721E-02	0.043249	0.272558	2.39949E-02	0.0613951	0.0049477	0.11663	108.2985	634.5347	652.5532	13.6977	
66	3.68831	1.6703E-02	0.037752	0.267070	2.17011E-02	0.0557498	0.0035717	0.11134	94.5501	624.1254	641.6563	13.6787	
67	3.68311	1.6685E-02	0.031944	0.261684	1.96702E-02	0.0495595	0.0024309	0.10529	79.9975	612.6339	629.7712	13.6597	
68	3.67514	1.6668E-02	0.025718	0.256188	1.73107E-02	0.0426298	0.0014794	0.10322	64.3965	599.6462	610.2529	13.6407	
69	3.66430	1.6651E-02	0.018831	0.250369	1.43748E-02	0.0345614	0.0010377	0.09954	47.1514	581.7401	592.2435	13.6219	
70	3.65093	1.6634E-02	0.010212	0.243630	9.50145E-03	0.0237315	0.0003179	0.02189	25.5716	561.6774	568.0473	13.6030	
71	12.13291	1.6513E-02	0.000706	0.223729	1.59855E-03	0.0029150	0.0000017	0.00158	1.7673	521.5089	521.5155	13.5839	
72	15.56918	1.6598E-02	0.	0.220937	0.	-0.0000000	0.	0.	0.	515.0000	515.0000	13.5650	

N	Y	G431	G432	G433	G434	G435	G436	G437	G438	G439	G4310	G4311	G4312
1	0.	0.03028	0.010737	0.021580	0.020954	0.023402	0.009158	0.017380	0.017449	0.000599	0.003522	0.001567	0.05328
2	0.	0.22042	0.010737	0.021580	0.030954	0.023402	0.009158	0.017380	0.017449	0.000599	0.003522	0.001567	0.05328
3	0.	0.15159	0.03020	0.010721	0.021550	0.030911	0.023170	0.009158	0.017377	0.000599	0.003517	0.001564	0.054128
4	0.	0.03015	0.010702	0.021511	0.030858	0.023131	0.009158	0.017377	0.017425	0.000599	0.003511	0.001561	0.054128
5	0.	0.03050	0.010681	0.021469	0.030798	0.023086	0.009158	0.017374	0.017368	0.000598	0.003508	0.001559	0.054128
6	0.	0.03117	0.030002	0.010657	0.021423	0.030733	0.023037	0.009094	0.017259	0.000591	0.003497	0.001557	0.054062
7	0.	0.03731	0.02995	0.010632	0.021373	0.030663	0.023116	0.009074	0.017221	0.000587	0.003489	0.001549	0.053972
8	0.	0.02701	0.02987	0.010606	0.021321	0.030590	0.023131	0.009051	0.017180	0.000584	0.003480	0.001545	0.053928
9	0.	0.02923	0.02980	0.010578	0.021266	0.030512	0.023073	0.009030	0.017134	0.000580	0.003473	0.001540	0.053872
10	0.	0.02974	0.02977	0.010549	0.021208	0.030431	0.023111	0.009006	0.017093	0.000576	0.003465	0.001535	0.053817
11	0.	0.02928	0.02961	0.010518	0.021148	0.030345	0.023088	0.008991	0.017046	0.000571	0.003454	0.001529	0.053757
12	0.	0.02834	0.02945	0.010486	0.021084	0.030255	0.023061	0.008975	0.016997	0.000567	0.003444	0.001524	0.053698
13	0.	0.02837	0.02944	0.010453	0.021017	0.030161	0.023011	0.008928	0.016935	0.000562	0.003434	0.001518	0.053639
14	0.	0.02846	0.02943	0.010417	0.020947	0.030063	0.022961	0.008870	0.016891	0.000559	0.003423	0.001512	0.053581
15	0.	0.02729	0.02924	0.010391	0.020874	0.029960	0.022961	0.008870	0.016838	0.000556	0.003411	0.001506	0.053524
16	0.	0.02019	0.02913	0.010362	0.020798	0.029853	0.022951	0.008838	0.016775	0.000553	0.003399	0.001499	0.053468
17	0.	0.02023	0.02911	0.010347	0.020719	0.029741	0.022947	0.008804	0.016713	0.000550	0.003387	0.001492	0.053412
18	0.	0.02045	0.02909	0.010320	0.020636	0.029624	0.022910	0.008772	0.016649	0.000547	0.003374	0.001485	0.053355
19	0.	0.02057	0.02907	0.010295	0.020549	0.029501	0.022910	0.008772	0.016582	0.000544	0.003360	0.001477	0.053298
20	0.	0.02137	0.02864	0.010171	0.020459	0.029374	0.022824	0.008699	0.016512	0.000540	0.003346	0.001469	0.053241
21	0.	0.02302	0.02851	0.010123	0.020364	0.029241	0.022724	0.008626	0.016438	0.000536	0.003331	0.001461	0.053184
22	0.	0.02313	0.02837	0.010074	0.020266	0.029102	0.022710	0.008600	0.016362	0.000532	0.003316	0.001453	0.053127
23	0.	0.02306	0.02822	0.010022	0.020163	0.028957	0.022713	0.008573	0.016282	0.000529	0.003300	0.001445	0.053070
24	0.	0.02302	0.02807	0.009968	0.020057	0.028807	0.022680	0.008536	0.016199	0.000526	0.003283	0.001438	0.053014
25	0.	0.02279	0.02791	0.009912	0.019945	0.028649	0.022653	0.008498	0.016113	0.000523	0.003266	0.001431	0.052958
26	0.	0.02274	0.02774	0.009858	0.019829	0.028486	0.022626	0.008461	0.016022	0.000520	0.003249	0.001424	0.052902
27	0.	0.02274	0.02757	0.009804	0.019709	0.028315	0.022633	0.008431	0.015929	0.000517	0.003232	0.001417	0.052846
28	0.	0.02237	0.02739	0.009730	0.019583	0.028150	0.022630	0.008399	0.015830	0.000514	0.003215	0.001410	0.052790
29	0.	0.02207	0.02720	0.009664	0.019452	0.027982	0.022627	0.008367	0.015728	0.000511	0.003198	0.001403	0.052734
30	0.	0.02173	0.02701	0.009595	0.019316	0.027760	0.022618	0.008329	0.015622	0.000508	0.003181	0.001396	0.052678
31	0.	0.02193	0.02681	0.009528	0.019178	0.027559	0.022608	0.008311	0.015511	0.000505	0.003164	0.001389	0.052622
32	0.	0.02024	0.02660	0.009469	0.019026	0.027350	0.022611	0.008311	0.015396	0.000505	0.003147	0.001382	0.052566
33	0.	0.02028	0.02638	0.009409	0.018872	0.027133	0.022634	0.008334	0.015276	0.000506	0.003129	0.001375	0.052510
34	0.	0.02193	0.02615	0.009352	0.018712	0.026906	0.022639	0.008339	0.015150	0.000506	0.003112	0.001368	0.052454
35	0.	0.02452	0.02591	0.009208	0.018546	0.026678	0.022603	0.008311	0.015020	0.000505	0.003095	0.001361	0.052398
36	0.	0.02106	0.02567	0.009121	0.018372	0.026425	0.022610	0.008311	0.014890	0.000504	0.003078	0.001354	0.052342
37	0.	0.02173	0.02541	0.009030	0.018192	0.026170	0.022610	0.008320	0.014763	0.000504	0.003061	0.001347	0.052286
38	0.	0.02253	0.02524	0.008936	0.018008	0.025904	0.022604	0.008320	0.014636	0.000504	0.003044	0.001340	0.052230
39	0.	0.02453	0.02504	0.008843	0.017809	0.025627	0.022627	0.008320	0.014509	0.000504	0.003027	0.001333	0.052174
40	0.	0.02156	0.02487	0.008748	0.017606	0.025339	0.022639	0.008320	0.014382	0.000504	0.003010	0.001326	0.052118
41	0.	0.02204	0.02459	0.008630	0.017395	0.025039	0.022631	0.008320	0.014255	0.000504	0.002993	0.001319	0.052062
42	0.	0.02362	0.02436	0.008520	0.017175	0.024727	0.022627	0.008320	0.014128	0.000504	0.002976	0.001312	0.052006
43	0.	0.02170	0.02408	0.008405	0.016946	0.024402	0.022630	0.008320	0.013994	0.000504	0.002959	0.001305	0.051950
44	0.	0.02607	0.02380	0.008285	0.016708	0.024064	0.022626	0.008320	0.013867	0.000504	0.002942	0.001298	0.051894
45	0.	0.02197	0.02351	0.008161	0.016460	0.023712	0.022631	0.008320	0.013740	0.000504	0.002925	0.001291	0.051838
46	0.	0.02000	0.02325	0.008032	0.016202	0.023345	0.022631	0.008320	0.013613	0.000504	0.002908	0.001284	0.051782
47	0.	0.02731	0.02320	0.007897	0.015932	0.023063	0.022631	0.008320	0.013486	0.000504	0.002891	0.001277	0.051726
48	0.	0.04037	0.02310	0.007757	0.015654	0.022763	0.022631	0.008320	0.013359	0.000504	0.002874	0.001270	0.051670
49	0.	0.03452	0.02140	0.007612	0.015363	0.022455	0.022631	0.008320	0.013232	0.000504	0.002857	0.001263	0.051614
50	0.	0.02462	0.02027	0.007460	0.015059	0.022118	0.022631	0.008320	0.013105	0.000504	0.002840	0.001256	0.051558

JET ANALYSIS PROGRAM A

MOLE FRACTIONS: STA 1332 Y= 25.00000 PRESSURE= 11.4300

N	Y	CAS1	CAS2	CAS3	CAS4	CAS5	CAS6	CAS7	CAS8	CAS9	CAS10	CAS11	CAS12
51	3.30745	0.002052	0.007302	0.014743	0.021264	0.015146	0.006320	0.012019	0.012077	0.004776	0.002836	0.001562	0.999491
52	3.45660	0.002006	0.007137	0.014412	0.020794	0.015074	0.006189	0.011757	0.011811	0.004672	0.002835	0.001532	0.999509
53	3.50621	0.001957	0.006945	0.014068	0.020309	0.014922	0.006044	0.011482	0.011532	0.004563	0.002830	0.001501	0.999526
54	3.60267	0.001806	0.006785	0.013709	0.019790	0.014638	0.005893	0.011195	0.011250	0.004449	0.002821	0.001469	0.999547
55	3.68842	0.001653	0.006638	0.013353	0.019253	0.014300	0.005735	0.010955	0.010995	0.004334	0.002811	0.001435	0.999579
56	3.83197	0.001746	0.006402	0.012940	0.018691	0.013920	0.005569	0.010581	0.010633	0.004205	0.002817	0.001399	0.911629
57	4.00594	0.001740	0.006197	0.012529	0.018102	0.013561	0.005395	0.010251	0.010302	0.004075	0.002806	0.001362	0.911629
58	4.20701	0.001640	0.005982	0.012097	0.017485	0.013294	0.005213	0.009904	0.009954	0.003931	0.002801	0.001325	0.911629
59	4.33626	0.001516	0.005757	0.011644	0.016835	0.012804	0.005021	0.009580	0.009636	0.003792	0.002794	0.001288	0.911629
60	4.51812	0.001379	0.005520	0.011168	0.016152	0.012286	0.004818	0.009156	0.009203	0.003640	0.002785	0.001250	0.923353
61	4.66247	0.001279	0.005270	0.010685	0.015430	0.011740	0.004604	0.008750	0.008795	0.003499	0.002776	0.001212	0.923353
62	4.86274	0.001193	0.005006	0.010134	0.014666	0.011142	0.004378	0.008320	0.008365	0.003358	0.002766	0.001174	0.923353
63	5.03725	0.001125	0.004726	0.009569	0.013853	0.010546	0.004157	0.007862	0.007907	0.003216	0.002756	0.001136	0.923353
64	5.28823	0.001052	0.004427	0.008967	0.012984	0.009848	0.003939	0.007373	0.007418	0.003074	0.002746	0.001098	0.923353
65	5.50622	0.001054	0.004106	0.008419	0.012052	0.009179	0.003701	0.006845	0.006890	0.002932	0.002736	0.001060	0.923353
66	5.75063	0.001054	0.003750	0.007816	0.011037	0.008409	0.003479	0.006401	0.006446	0.002790	0.002726	0.001022	0.923353
67	6.02311	0.000946	0.003375	0.006841	0.009918	0.007559	0.003296	0.005835	0.005880	0.002643	0.002716	0.000984	0.923353
68	6.40210	0.000825	0.002931	0.005984	0.008655	0.006597	0.003069	0.005422	0.005467	0.002499	0.002706	0.000946	0.923353
69	6.83544	0.000642	0.002532	0.004933	0.007157	0.005347	0.002814	0.004972	0.005017	0.002356	0.002696	0.000908	0.923353
70	7.49963	0.000480	0.001711	0.003473	0.005040	0.003944	0.002159	0.002869	0.002914	0.002214	0.002686	0.000870	0.923353
71	12.1291	0.000291	0.000325	0.000659	0.000257	0.000730	0.000284	0.000445	0.000450	0.000417	0.000411	0.000425	0.995471
72	15.56948	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	1.000000

• JET ANALYSIS PROGRAM •

FREE JET MIXING

• SUMMARY - STATION DATA - JET PROPERTIES •

N	X	B	YJ	UC	TC	TIC	PTC	TIC	YSD-IC	-J
1	0.02000	0.00139	1.01000	0.9230883	0.8831969	2.000000E-02	1.000000	1.000000	0.957372	1.000000
2	0.02000	0.12262	1.05210	0.9230883	0.8831974	1.998013E-02	0.931366	0.85123751	2.69577	1.032279
3	0.05000	0.23383	1.05310	0.9230887	0.8831991	1.995053E-02	0.931365	0.85123751	0.94197	1.055740
4	0.10000	0.26681	1.09321	0.9230912	0.8832055	1.990189E-02	0.931362	0.851137	0.89709	1.111151
5	0.20000	0.90284	1.41330	0.9231060	0.8832369	1.967412E-02	0.931359	0.855165	0.89709	1.277556
6	0.50000	0.70573	1.65577	0.9232719	0.8835365	1.955631E-02	0.931362	0.855518	0.91725	1.293371
7	1.00000	0.98105	1.87669	0.9233104	0.8846572	1.933247E-02	0.931364	0.855744	0.91725	1.352133
8	1.50000	1.02277	1.94198	0.9235054	0.8865701	1.931137E-02	0.931374	0.859180	0.92775	1.411472
9	2.00000	1.48816	1.71893	0.9266640	0.8862262	1.976441E-02	0.931412	0.866423	0.95572	1.503727
10	2.50000	1.63317	1.70911	0.9266642	0.8928175	2.077131E-02	0.931319	0.866423	0.95572	1.503727
11	2.77778	3.47771	1.31846	0.9300257	0.8925652	2.165742E-02	0.931323	0.866410	0.95572	1.503727
12	3.50000	4.46764	2.53162	1.0170166	0.9142416	4.223765E-02	1.031323	0.881314	1.013305	2.250456
13	4.00000	5.15355	2.57677	0.9472128	0.9374461	5.642397E-02	1.023545	0.931356	0.929395	2.604782
14	4.50000	5.28605	2.60802	0.9370509	0.8883841	7.800299E-02	0.966187	0.866410	0.95572	2.704482
15	5.00000	5.53141	2.74571	0.8700887	0.8350516	1.005151E-01	0.861759	0.794137	0.87272	2.975214
16	5.41514	5.73972	2.66926	0.9147058	0.7956526	1.133772E-01	0.771476	0.727240	0.87272	3.104544
17	5.55554	5.87406	2.93823	0.7971464	0.737069	1.163703E-01	0.751472	0.727240	0.87272	3.104544
18	6.00000	6.10174	3.05089	0.7866123	0.7491842	1.225277E-01	0.673842	0.672740	0.87272	3.104544
19	6.50000	6.34821	3.14960	0.6921284	0.7161113	1.251635E-01	0.586633	0.622130	0.87272	3.104544
20	7.00000	6.69543	3.46771	0.6460104	0.6879727	1.246919E-01	0.532252	0.561530	0.87272	3.104544
21	7.50000	6.93724	3.66458	0.6055044	0.6636395	1.223631E-01	0.477694	0.540121	0.87272	3.104544
22	8.00000	7.08107	3.83154	0.5688450	0.6423777	1.189536E-01	0.431519	0.515155	0.87272	3.104544
23	9.00000	7.60335	3.40017	0.5095075	0.6022250	1.109947E-01	0.357940	0.463740	0.87272	3.104544
24	10.00000	8.23557	4.11778	0.4400975	0.5754304	1.027072E-01	0.302200	0.421620	0.87272	3.104544
25	11.11111	8.95671	4.7816	0.4166573	0.5469849	9.430091E-02	0.254633	0.382383	0.87272	3.104544
26	12.00000	9.45067	4.72543	0.3833208	0.5265660	8.823774E-02	0.224714	0.352454	0.87272	3.104544
27	13.00000	10.11433	5.05717	0.3527285	0.5059291	8.213151E-02	0.192539	0.328393	0.87272	3.104544
28	14.00000	11.14681	5.59641	0.3036128	0.4717923	7.145800E-02	0.153476	0.284524	0.87272	3.104544
29	15.00000	12.32363	6.16182	0.2668052	0.4487386	6.275068E-02	0.123472	0.251749	0.87272	3.104544
30	16.00000	13.34531	6.67265	0.2313377	0.4272350	5.567199E-02	0.102516	0.225146	0.87272	3.104544
31	21.00000	18.53839	7.26920	0.2134653	0.4044320	4.966922E-02	0.086540	0.203354	0.87272	3.104544
32	22.22222	15.04922	7.53941	0.2011220	0.3944804	4.683377E-02	0.078317	0.191670	0.87272	3.104544
33	25.00000	11.06156	8.33028	0.1776360	0.3757367	4.105175E-02	0.063845	0.169628	0.87272	3.104544

EXECUTING PROGRAMS
TAPING 7

PSI	1	0.
1	0.001025	0.1025E-03
21	0.003240	0.003240
R		
1	0.026040	0.026040
21	0.009504	0.009504
G		
1	0.3108E-03	0.4729E-03
11	0.4080E-03	0.4080E-03
21	0.3108E-03	0.3108E-03
XS	0.020300	

PSI	1	0.
1	0.000325	0.277572
11	0.000650	0.304925
21	0.000975	0.408925
R		
1	0.001300	0.504111
11	0.001625	0.608244
21	0.001950	0.708244
G		
1	0.002275	0.808244
11	0.002600	0.908244
21	0.002925	1.008244
XS	0.020300	

PSI	1	0.
1	0.000325	0.277572
11	0.000650	0.304925
21	0.000975	0.408925
R		
1	0.001300	0.504111
11	0.001625	0.608244
21	0.001950	0.708244
G		
1	0.002275	0.808244
11	0.002600	0.908244
21	0.002925	1.008244
XS	0.020300	

ANALYTICAL MODEL OF EXHAUST PLUMES FROM AFTERBURNING ENGINES TO PREDICT REACTION OF CONTAMINANT EMISSIONS.
GENERAL ELECTRIC COMPANY AIRCRAFT ENGINE GROUP COMPUTED AT 23.18 HOURS ON 6/30/75 ELAPSED TIME 1428.07 SECONDS
J85-5 412.45 EFTC DEMONSTRATION TEST RVN 12-3

PROFILES AND CONTAMINANT RESIDUALS 0.060 FEET FROM NOZZLE EXIT									
RADIUS, FEET INNER	GAS FLOW PPS	VELOCITY FPS	FUEL/GAS	HIGH PART TEMP, DEGR	FUEL/GAS	LEAN PART TEMP, DEGR	PREDICTED INDICATION PT, PPM	TIME, SEC	
1.01331E-01	1.32285E 00	2.45700E 03	6.10925E-02	3.5231E 03	1.92585E-02	1.4775E 01	2.7726E 01	2.45471E 03	
2.07541E-01	4.29140E 00	2.55062E 03	6.42504E-02	3.5805E 03	1.88508E-02	1.4379E 03	2.8036E 01	2.45349E 01	
3.07541E-01	8.11282E 00	2.72916E 03	6.22283E-02	3.5861E 03	1.78185E-02	1.4142E 03	2.8003E 01	2.45156E 03	
4.07541E-01	1.17856E 01	2.88936E 03	6.11265E-02	3.5871E 03	1.6780E-02	1.3914E 03	2.8135E 01	2.45244E 03	
5.07541E-01	1.66597E 00	2.84346E 03	6.08265E-02	3.5862E 03	1.78429E-02	1.4094E 03	2.7848E 01	2.45494E 03	
6.07541E-01	3.56240E 00	2.63444E 03	6.14564E-02	3.5709E 03	1.76664E-02	1.4094E 03	2.8277E 01	2.45471E 03	
7.07541E-01	6.48240E 00	2.44571E 03	6.05810E-02	3.5844E 03	1.7724E-02	1.4094E 03	2.8527E 01	2.45442E 03	
8.07541E-01	8.48240E 00	1.74516E 03	5.18504E-02	3.5827E 03	1.5800E-02	1.3804E 03	2.6774E 01	2.45442E 03	
9.07541E-01	1.67450E 00	1.14648E 02	3.10824E-02	3.5744E 03	1.74413E-02	1.4094E 03	1.5357E 01	2.45442E 03	
1.00000E 00	1.74730E 00	2.84136E 00	1.10834E-02	1.28671E 03	5.5350E-03	1.1237E 03	1.5633E 01	1.10522E 03	
2.54744E 00	1.30749E 01	1.27099E 01	3.88460E-04	5.8453E 02	2.43244E-04	5.1064E 02	1.5611E 01	5.6553E 02	

AVERAGE GAS COMPOSITION UNDER FREE STREAM CONDITIONS

MASS FRACT RICH PART	FUEL/AIR RATIO	EMISSIONS INDICES, LB/KLB FUEL CO	HC NOX AS NO2	ANALYZER READINGS, PPM COLORIED)	HC AS CHN NOX
1.00000E-01	1.30734E-02	9.12740E 01	2.53166E 00	3.19498E 03	2.7130E 03
2.54744E-01	3.62708E-02	7.05249E 01	1.97708E 00	2.72179E 03	1.4204E 03
5.09314E-01	1.98426E-02	6.11170E 01	2.97874E 00	2.61129E 03	4.6746E 02
7.54494E-01	4.0834E-02	3.16800E 01	6.28831E-01	1.51046E 00	5.7429E 01
9.07710E-01	4.30914E-02	2.91272E 01	3.92831E 00	3.34001E 03	3.3102E 01
9.61614E-01	1.62117E-02	5.16266E 01	4.41074E 00	2.61315E 00	3.2223E 02
1.00000E-01	3.53131E-02	6.21605E 01	5.8148E 00	2.59757E 00	3.1555E 02
2.54744E-01	3.53131E-02	1.01271E 02	1.87952E 00	2.72965E 00	2.6834E 03
5.09314E-01	3.63351E-02	1.00981E 02	9.58426E 00	2.49950E 00	3.2970E 03
7.54494E-01	1.40243E 02	3.15617E 01	4.13877E 00	1.37283E 03	6.0733E 02
1.00000E 00	3.85647E-04	2.10142E 02	0.	0.11241E 00	0.3407E 01

AVERAGE GAS COMPOSITION AS MODIFIED BY SAMPLE PROBE

MASS FRACT RICH PART	FUEL/AIR RATIO	EMISSIONS INDICES, LB/KLB FUEL CO	HC NOX AS NO2	ANALYZER READINGS, PPM COLORIED)	HC AS CHN NOX
2.94014E-01	1.29052E-02	9.08266E 01	2.54361E 00	3.19491E 03	2.72731E 03
5.45873E-01	3.55873E-02	6.07156E 01	2.68837E 00	2.63622E 03	1.4552E 03
7.9673E-01	1.87056E-02	6.06216E 01	3.01915E 00	2.5515E 03	4.9513E 02
9.3657E-01	4.28419E-02	3.17245E 01	7.29930E-01	1.4370E 00	5.7429E 01
1.00000E-01	4.1775E-02	2.92318E 01	4.29755E-01	3.11798E 00	3.3995E 03
2.54744E-01	1.60271E-02	5.31316E 01	4.86663E 00	2.65990E 00	2.0373E 03
5.09314E-01	3.48665E-02	6.17976E 01	6.09466E 00	2.62167E 00	2.8934E 03
7.54494E-01	2.49125E-02	1.05286E 02	1.76119E 01	2.63091E 00	2.8709E 03
9.08266E-01	3.17832E-02	1.08926E 02	8.08246E 00	2.37751E 00	3.52997E 03
2.9257E-01	9.8741E-03	1.03210E 02	3.10930E 01	4.12757E 00	1.4069E 03
1.00000E 00	3.85647E-04	2.10142E 02	0.	0.11241E 00	0.3407E 01

INTEGRATED PROFILE AVERAGES (FREE STREAM CONDITIONS)

TOTAL FLOWS, PPS GAS MIX	EMISSIONS INDICES, LB/KLB FUEL CO	HC NOX AS NO2	CONTAMINANT FLOWS, PPS HC	NOX AS NO2
9.08259E 01	1.96103E 00	5.44751E 01	6.04221E 00	2.91116E 00
			1.08871E-01	1.34232E-02
				5.71110E-03

ANALYTICAL MODEL OF EXHAUST PLUMES FROM AFTERBURNING ENGINES TO PREDICT REACTION OF CONTAMINANT EMISSIONS.
 GENERAL ELECTRIC COMPANY AIRCRAFT ENGINE GROUP
 J85-5 WID-AN
 COMPUTED AT 21.19 HOURS ON 6/30/75
 ELAPSED TIME 1460.07 SECONDS
 RUN 32-3

PROFILES AND CONTAMINANT RESIDUALS 0.675 FEET FROM NOZZLE EXIT

RADIUS, FEET	GAS FLOW	VELOCITY	RICH PART	TEMP, DEGR	FUEL/GAS	LEAN PART	PREDICTED INDICATION
INNER	OUTER	PPS	FUEL/GAS	TEMP, DEGR	FUEL/GAS	TEMP, DEGR	PT, PSIA
0.00000E+00	1.24325E+00	2.45023E+00	4.89188E-02	3.03871E+03	1.90440E-02	1.47248E+03	2.46301E+01
0.00000E+00	2.06302E+00	2.60031E+00	4.94992E-02	3.03256E+03	1.84567E-02	1.43779E+03	2.46301E+01
0.00000E+00	3.23321E+00	2.77105E+00	5.17508E-02	3.11282E+03	1.78652E-02	1.41208E+03	2.46301E+01
0.00000E+00	4.91850E+00	2.90191E+00	5.35835E-02	3.18185E+03	1.72532E-02	1.38748E+03	2.46301E+01
0.00000E+00	6.60672E+00	2.74228E+00	5.09327E-02	3.08386E+03	1.71907E-02	1.36741E+03	2.46301E+01
0.00000E+00	8.29476E+00	2.35521E+00	4.31667E-02	2.79596E+03	1.63080E-02	1.40173E+03	2.46301E+01
0.00000E+00	1.00870E+01	1.78632E+00	3.54802E-02	1.96753E+03	1.50590E-02	1.50794E+03	2.46301E+01
0.00000E+00	1.18632E+01	1.40069E+00	1.18632E+00	1.26233E+03	1.57098E-02	1.63743E+03	2.46301E+01
0.00000E+00	1.36383E+01	1.22540E+00	1.18632E+00	7.98390E+02	1.56740E-02	1.64640E+03	2.46301E+01
0.00000E+00	1.54135E+01	1.09216E+00	1.33458E-02	1.45533E+03	1.56740E-02	1.64640E+03	2.46301E+01
0.00000E+00	1.71886E+01	1.00870E+00	1.12858E+00	5.49354E+02	0.00000E+00	5.14605E+02	1.36309E+01
0.00000E+00	1.89638E+01	1.00870E+00	1.12858E+00	5.49354E+02	0.00000E+00	5.14605E+02	1.36309E+01

AVERAGE GAS COMPOSITION UNDER FREE STREAM CONDITIONS

WASS FRACT	FUEL/AIR	EMISSIONS INDICES	LB/KLB FUEL	NOX AS NO2	COLORIED	ANALYZER READINGS, PPMV	NOX
RICH PART	RATIO	CO	HC	NOX AS NO2	COLORIED	HC AS CMV	NOX
0.00000E+00	1.24325E+00	6.56005E+01	2.71141E+01	2.54509E+00	2.37635E+03	1.83922E+03	5.33094E+01
0.00000E+00	2.06302E+00	5.60766E+01	1.37711E+01	2.72164E+00	2.22823E+03	1.00112E+03	6.03185E+01
0.00000E+00	3.23321E+00	4.69000E+01	6.00333E+00	2.96864E+00	2.00965E+03	7.76101E+02	7.15413E+01
0.00000E+00	4.91850E+00	3.02137E+01	1.01146E+00	3.08378E+00	1.40520E+03	6.63244E+01	8.00776E+01
0.00000E+00	6.60672E+00	3.14109E+01	1.01841E+00	2.95124E+00	1.41616E+03	1.42260E+02	7.03876E+01
0.00000E+00	8.29476E+00	4.35066E+01	1.29019E+00	2.65671E+00	1.45052E+03	3.31534E+02	5.48464E+01
0.00000E+00	1.00870E+01	6.11242E+01	7.73845E+00	2.71790E+00	1.81780E+03	3.89003E+02	3.78859E+01
0.00000E+00	1.18632E+01	6.75955E+01	7.44593E+00	2.75266E+00	1.71168E+03	1.98205E+02	2.19874E+01
0.00000E+00	1.36383E+01	1.03678E+02	7.55918E+00	2.75200E+00	4.50108E+02	2.2712E+01	7.50399E+00
0.00000E+00	1.54135E+01	6.71394E+01	8.26074E+00	2.74309E+00	1.23587E+03	2.24053E+02	2.30664E+01
0.00000E+00	1.71886E+01	1.18024E+02	4.91667E+00	2.91109E+00	5.75653E+01	4.75937E+00	8.59179E+01

AVERAGE GAS COMPOSITION AS MODIFIED BY SAMPLE PROBE

WASS FRACT	FUEL/AIR	EMISSIONS INDICES	LB/KLB FUEL	NOX AS NO2	COLORIED	ANALYZER READINGS, PPMV	NOX
RICH PART	RATIO	CO	HC	NOX AS NO2	COLORIED	HC AS CMV	NOX
0.00000E+00	1.24325E+00	6.50813E+01	2.40863E+01	2.61194E+00	2.32694E+03	1.98012E+03	5.33034E+01
0.00000E+00	2.06302E+00	5.65097E+01	1.45928E+01	2.75170E+00	2.17021E+03	1.00315E+03	5.99664E+01
0.00000E+00	3.23321E+00	4.66406E+01	5.15449E+00	2.94688E+00	1.95310E+03	9.9213E+02	7.07086E+01
0.00000E+00	4.91850E+00	3.02728E+01	1.11711E+00	3.10822E+00	1.37295E+03	3.30352E+01	7.88988E+01
0.00000E+00	6.60672E+00	3.54135E+01	1.07530E+00	2.98011E+00	1.36374E+03	1.51237E+02	6.95594E+01
0.00000E+00	8.29476E+00	4.33910E+01	5.47469E+00	2.87548E+00	1.43161E+03	3.59248E+02	5.42111E+01
0.00000E+00	1.00870E+01	6.93596E+01	7.62198E+00	2.76768E+00	1.62666E+03	3.41729E+02	3.78295E+01
0.00000E+00	1.18632E+01	8.60338E+01	8.00087E+00	2.80140E+00	1.15747E+03	2.10987E+02	2.23997E+01
0.00000E+00	1.36383E+01	1.01049E+02	3.53546E+00	2.81572E+00	4.55952E+02	3.16570E+01	7.69161E+00
0.00000E+00	1.54135E+01	6.70257E+01	8.30774E+00	2.74702E+00	1.23454E+03	2.29820E+02	2.31271E+01
0.00000E+00	1.71886E+01	1.18924E+02	4.91667E+00	2.91109E+00	5.75653E+01	4.75937E+00	8.59179E+01

INTEGRATED PROFILE AVERAGES (FREE STREAM CONDITIONS)

TOTAL FLOW, PPS	EMISSIONS INDICES	LB/KLB FUEL	NOX AS NO2	CONTAMINANT FLOWS, PPS	HC	NOX AS NO2
GAS FLOW	CO	HC	NOX AS NO2	CO	HC	NOX AS NO2
0.00259E+01	1.96183E+00	4.91388E+01	3.06798E+00	2.91555E+00	9.60017E+02	9.94152E+03
						5.71196E+03

ANALYTICAL MODEL OF EXHAUST PLUMES FROM AFTERBURNING ENGINES TO PREDICT REACTION OF CONTAMINANT EMISSIONS.
GENERAL ELECTRIC COMPANY AIRCRAFT ENGINE GROUP
JAS-5-122A8

COMPUTED AT 23.27 HOURS ON 6/30/75
 ELAPSED TIME 1725.92 SECONDS
 JUNE 30-75

PROFILES AND CONTAMINANT RESIDUALS 7.500 FEET FROM NOZZLE EXIT

RADIUS, FEET INNER	GAS FLOW PPS	VELOCITY FPS	RICH PART		LEAN PART		PREDICTED INDICATION	
			FUEL/GAS	TEMP. DEGR	FUEL/GAS	TEMP. DEGR	PT. PSIA	TT. DEGR
0.10301F-01	1.79137E 00	2.06986E 03	2.91191E-02	1.69071E 03	1.61971E-02	1.51142E 03	2.51249E 01	1.49923E 03
0.20601F-01	6.63540E 00	1.97406E 03	2.50845E-02	1.45024E 03	1.59174E-02	1.57471E 03	2.41295E 01	1.45151E 03
0.30901F-01	1.55349E 01	1.72194E 03	2.02944E-02	1.73511E 03	1.51121E-02	1.58743E 03	2.17503E 01	1.72622E 03
0.41201F-01	3.08729E 01	1.52828E 03	1.50824E-02	1.54003E 03	1.50421E-02	1.6821E 03	1.87524E 01	1.54755E 03
0.51501F-01	4.94742E 01	1.32828E 03	1.20349E-02	1.32907E 03	1.50305E-02	1.71592E 03	1.64274E 01	1.34744E 03
0.61801F-01	7.17161E-01	9.35591E 02	9.68491E-03	1.20177E 03	1.50349E-02	1.73677E 03	1.54923E 01	1.22132E 03
0.72101F-01	1.03617E 00	7.18591E 02	8.08148E-03	1.04450E 03	1.57157E-02	1.78731E 03	1.44923E 01	1.19911E 03
0.82401F-01	1.29617E 00	5.39137E 02	7.18670E-03	1.04450E 03	1.57157E-02	1.78731E 03	1.44923E 01	1.19911E 03
0.92701F-01	1.61155E 00	2.72776E 02	3.97275E-03	7.77371E 02	1.57157E-02	1.78731E 03	1.44923E 01	1.19911E 03
1.03001F-01	1.77495E 00	1.09242E 02	3.97275E-03	7.77371E 02	1.57157E-02	1.78731E 03	1.44923E 01	1.19911E 03
1.13301F-01	1.99718E 00	3.04500E 01	8.11048E-03	1.11749E 03	1.57157E-02	1.78731E 03	1.44923E 01	1.19911E 03
1.23601F-01	2.01718E 00	2.60279E 02	2.91029E-04	5.37042E 02	0.	5.18213E 02	1.42537E 01	5.37042E 02

AVERAGE GAS COMPOSITION UNDER FREE STREAM CONDITIONS

MASS FRACT RICH PART	FUEL/AIR RATIO	EMISSIONS INDICES, LB/KLB FUEL		ANALYZER READINGS, PPMV	
		CO	HC	CO (CORR)	HC AS CHN
0.10301F-01	2.92148E-02	2.59299E 01	1.33819E 00	2.90828E 00	6.59012E 02
0.20601F-01	2.91736E-02	2.59299E 01	1.33819E 00	2.90828E 00	6.59012E 02
0.30901F-01	2.91736E-02	2.59299E 01	1.33819E 00	2.90828E 00	6.59012E 02
0.41201F-01	2.91736E-02	2.59299E 01	1.33819E 00	2.90828E 00	6.59012E 02
0.51501F-01	2.91736E-02	2.59299E 01	1.33819E 00	2.90828E 00	6.59012E 02
0.61801F-01	2.91736E-02	2.59299E 01	1.33819E 00	2.90828E 00	6.59012E 02
0.72101F-01	2.91736E-02	2.59299E 01	1.33819E 00	2.90828E 00	6.59012E 02
0.82401F-01	2.91736E-02	2.59299E 01	1.33819E 00	2.90828E 00	6.59012E 02
0.92701F-01	2.91736E-02	2.59299E 01	1.33819E 00	2.90828E 00	6.59012E 02
1.03001F-01	2.91736E-02	2.59299E 01	1.33819E 00	2.90828E 00	6.59012E 02
1.13301F-01	2.91736E-02	2.59299E 01	1.33819E 00	2.90828E 00	6.59012E 02
1.23601F-01	2.91736E-02	2.59299E 01	1.33819E 00	2.90828E 00	6.59012E 02

AVERAGE GAS COMPOSITION AS MODIFIED BY SAMPLE PROBE

MASS FRACT RICH PART	FUEL/AIR RATIO	EMISSIONS INDICES, LB/KLB FUEL		ANALYZER READINGS, PPMV	
		CO	HC	CO (CORR)	HC AS CHN
0.10301F-01	2.92148E-02	2.59299E 01	1.33819E 00	2.90828E 00	6.59012E 02
0.20601F-01	2.91736E-02	2.59299E 01	1.33819E 00	2.90828E 00	6.59012E 02
0.30901F-01	2.91736E-02	2.59299E 01	1.33819E 00	2.90828E 00	6.59012E 02
0.41201F-01	2.91736E-02	2.59299E 01	1.33819E 00	2.90828E 00	6.59012E 02
0.51501F-01	2.91736E-02	2.59299E 01	1.33819E 00	2.90828E 00	6.59012E 02
0.61801F-01	2.91736E-02	2.59299E 01	1.33819E 00	2.90828E 00	6.59012E 02
0.72101F-01	2.91736E-02	2.59299E 01	1.33819E 00	2.90828E 00	6.59012E 02
0.82401F-01	2.91736E-02	2.59299E 01	1.33819E 00	2.90828E 00	6.59012E 02
0.92701F-01	2.91736E-02	2.59299E 01	1.33819E 00	2.90828E 00	6.59012E 02
1.03001F-01	2.91736E-02	2.59299E 01	1.33819E 00	2.90828E 00	6.59012E 02
1.13301F-01	2.91736E-02	2.59299E 01	1.33819E 00	2.90828E 00	6.59012E 02
1.23601F-01	2.91736E-02	2.59299E 01	1.33819E 00	2.90828E 00	6.59012E 02

INTEGRATED PROFILE AVERAGES (FREE STREAM CONDITIONS)

TOTAL FLOW, PPS		EMISSIONS INDICES, LB/KLB FUEL		CONTAMINANT FLOWS, PPS	
GAS "1"		CO	HC	CO	HC
1.79757E 02	1.96103E 00	3.20694E 01	2.25402E 00	2.91172E 00	6.29098E-02
				9.42357E-03	5.71230E-03

ANALYTICAL MODEL OF EXHAUST PLUMES FROM AFTERBURNING ENGINES TO PREDICT REACTION OF CONTAMINANT EMISSIONS.
 GENERAL ELECTRIC COMPANY AIRCRAFT ENGINE GROUP COMPUTED AT 21.31 HOURS ON 6/30/75 ELAPSED TIME 1655.76 SECONDS
 J05-S 110.43 EFC DEMONSTRATION TEST PWY 32-3

PROFILES AND CONTAMINANT RESIDUALS 15,000 FEET FROM NOZZLE EXIT

RADIUS, FEET INNER	GAS FLOW PPS	VELOCITY FPS	FUEL/GAS	TEMP, DEGR	LEAN PART FUEL/GAS	TEMP, DEGR	PREDICTED INDICATION	
							PT, PSIA	TT, DEGR
1.04850E-01	1.57720E-00	1.06030E-03	1.17037E-02	1.26590E-03	1.93972E-02	1.58931E-03	1.74395E-01	1.30643E-03
1.04850E-01	1.30640E-01	1.01970E-03	1.13743E-02	1.26934E-03	1.92135E-02	1.58286E-03	1.71848E-01	1.24144E-03
1.24090E-01	7.23900E-01	9.12470E-02	1.04972E-02	1.26334E-03	1.37052E-02	1.55866E-03	1.65266E-01	1.23442E-03
7.23940E-01	1.12930E-00	5.04900E-01	8.92540E-03	1.13533E-03	1.26589E-02	1.50294E-03	1.55570E-01	1.15756E-03
1.12940E-00	1.45040E-00	5.03610E-01	7.14440E-03	1.02660E-03	1.14322E-02	1.42673E-03	1.47342E-01	1.04111E-03
1.45040E-00	1.59040E-00	2.00720E-01	5.99640E-03	9.51840E-02	1.07355E-02	1.30433E-03	1.43531E-01	9.5527E-02
1.59040E-00	1.91310E-00	3.13750E-02	5.00623E-03	8.44040E-02	1.01743E-02	1.54231E-03	1.43701E-01	8.73742E-02
1.91310E-00	2.41320E-00	5.77500E-01	4.18430E-03	7.22340E-02	9.36395E-03	1.28511E-03	1.37766E-01	7.74522E-02
2.41320E-00	2.66780E-00	1.10750E-01	2.96723E-03	6.72420E-02	9.35275E-03	1.25546E-03	1.36612E-01	6.72422E-02
2.66780E-00	2.93390E-00	1.17750E-01	1.94723E-03	6.21040E-02	9.64035E-03	1.14443E-02	1.36349E-01	6.21041E-02
2.93390E-00	3.32251E-00	1.18745E-01	9.27790E-04	5.86070E-02	1.44440E-02	5.14940E-02	1.36310E-01	5.14940E-02

AVERAGE GAS COMPOSITION UNDER FREE STREAM CONDITIONS

MASS FRACT RICH FRACT	FUEL/AIR RATIO	EMISSIONS INDICES, LB/KLR FUEL CO	HC	COLOURED)	ANALYZER READINGS, PPMV	
					HC AS CHN	NOX
9.01125E-01	1.19815E-02	3.07662E-01	2.13540E-00	2.91221E-00	3.93507E-02	5.19525E-01
9.03117E-01	1.16466E-02	3.08602E-01	2.14536E-00	2.91217E-00	3.75040E-02	5.07618E-01
9.04411E-01	1.07542E-02	3.11152E-01	2.17330E-00	2.91207E-00	3.47680E-02	4.75254E-01
9.06930E-01	1.09520E-03	3.15751E-01	2.24010E-00	2.91180E-00	2.99690E-02	4.14441E-01
9.74097E-01	7.32661E-03	3.20368E-01	2.27537E-00	2.91167E-00	2.83030E-02	3.40191E-01
9.73650E-01	6.15448E-03	3.25957E-01	2.30446E-00	2.91144E-00	2.65392E-02	2.89548E-01
9.64750E-01	5.12620E-03	3.25463E-01	2.35235E-00	2.91141E-00	1.72370E-02	2.44573E-01
9.74097E-01	3.45496E-03	3.29251E-01	2.37520E-00	2.91120E-00	1.17325E-02	1.68152E-01
1.00000E-01	2.07739E-03	3.30715E-01	2.39197E-00	2.91112E-00	7.07590E-02	1.01754E-01
1.00000E-02	1.39291E-03	3.31441E-01	2.40020E-00	2.91107E-00	4.75158E-01	6.86448E-00
1.00000E-00	9.31159E-04	3.31656E-01	2.40307E-00	2.91106E-00	3.17738E-01	4.59675E-00

AVERAGE GAS COMPOSITION AS MODIFIED BY SAMPLE PRUNE

MASS FRACT RICH FRACT	FUEL/AIR RATIO	EMISSIONS INDICES, LB/KLR FUEL CO	HC	COLOURED)	ANALYZER READINGS, PPMV	
					HC AS CHN	NOX
9.58719E-01	1.19815E-02	3.07662E-01	2.13540E-00	2.91207E-00	3.93507E-02	5.19525E-01
9.58719E-01	1.16466E-02	3.08602E-01	2.14536E-00	2.91207E-00	3.75040E-02	5.07618E-01
9.58719E-01	1.07542E-02	3.11152E-01	2.17330E-00	2.91207E-00	3.47680E-02	4.75254E-01
9.58719E-01	1.09520E-03	3.15751E-01	2.24010E-00	2.91180E-00	2.99690E-02	4.14441E-01
9.74097E-01	7.32661E-03	3.20368E-01	2.27537E-00	2.91167E-00	2.83030E-02	3.40191E-01
9.74097E-01	6.15448E-03	3.25957E-01	2.30446E-00	2.91144E-00	2.65392E-02	2.89548E-01
9.64750E-01	5.12620E-03	3.25463E-01	2.35235E-00	2.91141E-00	1.72370E-02	2.44573E-01
9.74097E-01	3.45496E-03	3.29251E-01	2.37520E-00	2.91120E-00	1.17325E-02	1.68152E-01
1.00000E-01	2.07739E-03	3.30715E-01	2.39197E-00	2.91112E-00	7.07590E-02	1.01754E-01
1.00000E-02	1.39291E-03	3.31441E-01	2.40020E-00	2.91107E-00	4.75158E-01	6.86448E-00
1.00000E-00	9.31159E-04	3.31656E-01	2.40307E-00	2.91106E-00	3.17738E-01	4.59675E-00

INTEGRATED PROFILE AVERAGES (FREE STREAM CONDITIONS)

TOTAL FLOW, PPS GAS MIX	EMISSIONS INDICES, LB/KLR FUEL CO	HC	CO	CONTAMINANT FLOWS, PPS	
				HC	NOX AS NO2
3.05143E-02	1.06183E-00	3.14840E-01	2.25407E-00	2.91114E-00	6.24619E-02
				4.42366E-03	5.71232E-03

3.4 Helpful Hints

At values of x/dj specified by variable X in Namelist \$A for JETINP, JETMIX not only prints its mixing profiles, but also writes profile data on a file for transmission to links SPALDG and EMIS. Suggested values for X are: 0.0, 0.02, 0.05, 0.1, 0.2, 0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5, 5.0, 5.5, 6.0, 6.5, 7.0, 7.5, 8.0, 9.0, 10.0, 11.0, 12.0, 13.0, 15.0, 17.0, 19.0, 21.0, 23.0, and 25.0. Jet diameter (dj) is twice the value input for RADJ or AMBT(5) in Namelist \$TSTDAT. Emissions print stations need not correspond to JETMIX print stations. Emissions reactions should be frozen by $x/dj = 25.0$, but several intermediate stations should be printed to verify this.

The recommended value for CXPC in Namelist \$A for JETINP is 0.15. This value is chosen as a compromise between computational accuracy and reasonable computer time. Occasionally, a malfunction may result in JETMIX due to addition of excessively large streamtubes at the outer boundary, prior to reaching the end of the potential core. If this occurs, try $CXPC = 0.015$ and restart JETMIX from the last print station having reasonable streamtube size ($\Delta\psi$) at the outer boundary.

To restart JETMIX, delete the input for PREJET (Namelist \$TSTDAT and \$A for JETPRF and the preceding formatted cards), and set the variable RESTRT in \$A for JETINP equal to the value of X at which restart is to occur.

The PLUMOD version of GCKP has more ability to recover from computational malfunctions than the standard NASA edition; even so, problems may arise. The initial conditions for each kinetics calculation are recorded on scratch file 40, and should be dumped automatically in the event of a failure in GCKP. These data are sufficient to permit running the troublesome calculation on a stand-alone version of GCKP, if available. It is usually possible to circumvent the problem by adjustment of the step size controls in Namelist \$PROB.

SECTION 4.0

SOURCE LISTINGS

Listings of the FORTRAN source decks of the various subprograms follow, in alphabetical order by deck label. To find a deck containing a given entry name (SYMDEF), the user may refer to Table 1. The locations of the various decks in the program overlay structure is given in Sections 2.1 and 2.3 above.

C	ADJMC	MW OF RAW FUEL// ADJUST C2H4 THERMO COEF	ADJMC001
		SUBROUTINE ADJMC	ADJMC002
		COMMON /INDATA/ N,HF,WAR,T2,BETA,T25,FAR5,EINOC,P0,	ADJMC003
	*	RCO(11),RCO2(11),RHC(11),RNOX(11),	ADJMC004
	*	PT(11),PS(11),HLOC(11),OCO(11)	ADJMC005
	REAL	N	ADJMC006
	COMMON /TCOF	/ TC(7,2,25),TLCH,TMID,THI	ADJMC007
	COMMON /SNMW	/ ALSP(2,75), ALMW(75)	ADJMC008
C			ADJMC009
C		CALCULATE MW OF FUEL	ADJMC010
C		ALMW(16)= 10.*(12.01+1.008*N)	ADJMC011
C			ADJMC012
C		ADJUST C2H4 THERMO COEFFICIENT	ADJMC013
C			ADJMC014
			ADJMC015
	2	HV = (184686.04+37977.7*N)/(11.91868+N)	ADJMC016
		ADD = (HV-20275.+155.)*28.054/1.98596	ADJMC017
		TC(6,1,16)= TC(6,1,16)+ADD	ADJMC018
		TC(6,2,16)= TC(6,2,16)+ADD	ADJMC019
C			ADJMC020
	3	RETURN	ADJMC021
		END	ADJMC022

CALFA2	SOLVE FOR MASS TRANSFER MATRIX	ALFA2001
C		ALFA2002
	SUBROUTINE ALFA2 (N)	ALFA2003
C		ALFA2004
	COMMON/CXFLT/W(12,12),WHAT(13,12),ALFA(12,12)	ALFA2005
	COMMON/CNWNET/DUM1(100),WFCUM(12),DUM2(100)	ALFA2006
	COMMON/CMASS /WT(12),WTHAT(12),DUM3(24)	ALFA2007
C		ALFA2008
	DIMENSION WF(11)	ALFA2009
C		ALFA2010
	NAMEI IST /RADALF/ W,ALFA,WHAT	ALFA2011
C		ALFA2012
	N IS NUMBER OF FUEL-BEARING TUBES	ALFA2013
C		ALFA2014
	W(K,J) IS FLOW OF SPECIES K IN TUBE J AT OLD STATION	ALFA2015
C		ALFA2016
	WHAT(K,I) IS FLOW OF SPECIES K IN TUBE I AT NEW STATION	ALFA2017
C		ALFA2018
	ALFA(J,I) IS FRACTION OF FLOW IN TUBE J AT OLD STATION	ALFA2019
C		ALFA2020
	THAT IS TRANSFERRED TO TUBE I AT NEW STATION	ALFA2021
C		ALFA2022
	WF(J) IS FUEL FLOW IN TUBE J (CONSTANT BY TUBE DEFINITION)	ALFA2023
C		ALFA2024
	WFCUM IS CUMULATIVE FUEL FLOW	ALFA2025
C		ALFA2026
	IF(N.LT.2)CALL ERROR	ALFA2027
	N1=N-1	ALFA2028
	M=N+1	ALFA2029
	CALL SETM(1,0,0,ALFA,144)	ALFA2030
C		ALFA2031
	GET FUEL FLOW IN EACH TUBE	ALFA2032
C		ALFA2033
	DO 5 J=1,N	ALFA2034
	5 WF(J)=WFCUM(J+1)-WFCUM(J)	ALFA2035
C		ALFA2036
	TUBE 1	ALFA2037
C		ALFA2038
	RFP1=WF(2)/WF(1)	ALFA2039
	ALFA(2,1)=(W(1,1)-WHAT(1,1))/(W(1,1)*RFP1-W(1,2))	ALFA2040
	IF (WHAT(M,1)/WTHAT(1).GE.0.01)	ALFA2041
	ALFA(2,1)=(WTHAT(1)-WT(1))/(WT(2)-RFP1*WT(1))	ALFA2042
	ALFA(2,1)=AMINI(ALFA(2,1),1.0,1.0/RFP1)	ALFA2043
	IF (ALFA(2,1).LT.0.0) ALFA(2,1)=0.0	ALFA2044
	ALFA(1,2)=AMINI(1.0,RFP1*ALFA(2,1))	ALFA2045
	ALFA(1,1)=1.0-ALFA(1,2)	ALFA2046
C		ALFA2047
	TUBES 2 THRU (N-1)	ALFA2048
C		ALFA2049
	IF(N.EQ.2)GO TO 15	ALFA2050
	DO 10 J=2,N1	
	RFP1=WF(J+1)/WF(J)	
	RFM1=WF(J-1)/WF(J)	
	ALFA(J+1,J)=(W(J,J)-WHAT(J,J)+(W(J,J-1)-RFM1*W(J,J))*ALFA(J-1,J))	
	1/(RFP1*W(J,J)-W(J,J+1))	
	IF (WHAT(M,J)/WTHAT(J).GE.0.01)	

1	ALFA(J+1,J)=(WTHAT(J)-WT(J)-(WT(J-1)-RFP1*WT(J))*ALFA(J-1,J))	ALFA2051
1	/(WT(J+1)-RFP1*WT(J))	ALFA2052
	ATFMP=1.0-ALFA(J,J-1)	ALFA2053
	ALFA(J+1,J)=AMIN1(ALFA(J+1,J),1.0,ATFMP/RFP1)	ALFA2054
	IF (ALFA(J+1,J).LT.0.0) ALFA(J+1,J)=0.0	ALFA2055
	ALFA(J,J+1)=AMIN1(ATEMP,RFP1*ALFA(J+1,J))	ALFA2056
10	ALFA(J,J)=AMAX1(0.0,ATEMP-ALFA(J,J+1))	ALFA2057
C		ALFA2058
C	TUBE N	ALFA2059
C		ALFA2060
	15 ALFA(N,N)=1.0-ALFA(N,N1)	ALFA2061
C		ALFA2062
C	AIR ENTRAINMENT	ALFA2063
C		ALFA2064
	ALFA(M,N)=1.0	ALFA2065
C		ALFA2066
C	FINAL CHECK FOR OUT-OF-BOUNDS ALFA'S	ALFA2067
C		ALFA2068
	ALFMIN= 0.	ALFA2069
	ALFMAX= 1.	ALFA2070
	DO 17 I=1,M	ALFA2071
	DO 17 J=1,M	ALFA2072
	ALFMIN= AMIN1(ALFMIN,ALFA(J,I))	ALFA2073
17	ALFMAX= AMAX1(ALFMAX,ALFA(J,I))	ALFA2074
	IF((ALFMIN.GE.0.) .AND. (ALFMAX.LE.1.)) GO TO 70	ALFA2075
	WRITE (6,BADALF)	ALFA2076
C	KILL THE MIXING ON THIS STEP	ALFA2077
	CALL SETM(1,0.,ALFA,100)	ALFA2078
	DO 18 J=1,M	ALFA2079
	CALL MOVF(1,W(1,J),WHAT(1,J),12,1)	ALFA2080
18	ALFA(J,J)= 1.	ALFA2081
70	RETURN	ALFA2082
C		ALFA2083
	END	ALFA2084

CAUT00	AUTOMATIC ELIMINATION FROM ERROR CONSIDERATIONS	AUT00001
C	OF SPECIES WITH NON-REPRESENTATIVE ERRORS	AUT00002
	SUBROUTINE AUTO	AUT00003
C		AUT00004
C	AUTOMATIC ELIMINATION FROM ERROR CONSIDERATIONS OF SPECIES WITH	AUT00005
C	NON-REPRESENTATIVE ERRORS	AUT00006
C		AUT00007
	REAL MEDIAN	AUT00008
C		AUT00009
C	DIMENSION ERROR(25),EE(25)	AUT00010
C		AUT00011
	COMMON/CONN/DUM1(33),LS,LSP3,NEXT	AUT00012
	COMMON/PORF/DUM2(84),E(28)	AUT00013
	COMMON/SKIP/NEGL(25),I1,I2,I7	AUT00014
C		AUT00015
	EQUIVALENCE (EE(1),E(4))	AUT00016
C		AUT00017
	I2 = 0	AUT00018
	M = LS - I1	AUT00019
	IF (M .LE. 3) RETURN	AUT00020
C		AUT00021
	K = 0	AUT00022
	DO 3 I=1,LS	AUT00023
	IF (I1 .EQ. 0) GO TO 2	AUT00024
	DO 1 J=1,I1	AUT00025
	IF (I .EQ. NEGL(J)) GO TO 3	AUT00026
	1 CONTINUE	AUT00027
	2 K = K + 1	AUT00028
	ERROR(K) = EE(I)	AUT00029
	3 CONTINUE	AUT00030
C		AUT00031
	N = M/2 + 1	AUT00032
C		AUT00033
	DO 5 NN=1,N	AUT00034
	EMX = ERROR(NN)	AUT00035
	I = NN	AUT00036
	NI = NN + 1	AUT00037
	DO 4 II=NI,M	AUT00038
	IF (ERROR(II) .LE. EMX) GO TO 4	AUT00039
	EMX = ERROR(II)	AUT00040
	I = II	AUT00041
	4 CONTINUE	AUT00042
	ERROR(I) = ERROR(NN)	AUT00043
	ERROR(NN) = EMX	AUT00044
	5 CONTINUE	AUT00045
C		AUT00046
	MEDIAN = ERROR(N)	AUT00047
	CUTOFF = 15.*MEDIAN	AUT00048
C		AUT00049
	DO 8 I=1,LS	AUT00050

IF (I1 .EQ. 0) GO TO 7	AUTO0051
DO 6 J=1,I1	AUTO0052
IF (I .EQ. NEGL(J)) GO TO 8	AUTO0053
6 CONTINUE	AUTO0054
7 IF (FE(I) .LE. CUTOFF) GO TO 8	AUTO0055
I2 = I2 + 1	AUTO0056
K = I1 + I2	AUTO0057
NEGL(K) = I	AUTO0058
8 CONTINUE	AUTO0059
IT = I1 + I2	AUTO0060
C	AUTO0061
RETURN	AUTO0062
END	AUTO0063

CBLCK	-BLCK-	BLOCK DATA (SP/MW)	
C			BLCK0001
	BLOCK DATA		BLCK0002
C			BLCK0003
*	SPECIES NAMES AND MOLECULAR WEIGHTS		BLCK0004
C			BLCK0005
	COMMON / SN4W / ALSP(2,75), ALMW(75)		BLCK0006
C			BLCK0007
	DATA	AL SP/	BLCK0008
1	6HH	,6H	BLCK0009
2	6H2	,6H	BLCK0010
3	6HCO	,6H	BLCK0011
4	6HAR	,6H	BLCK0012
5	6HHN2	,6H	BLCK0013
6	6HFUEL	,6H	BLCK0014
7	118*0./		BLCK0015
C			BLCK0016
	DATA	AL MW/	BLCK0017
1	1,008, 16,000, 2,016, 32,000, 17,008, 18,016,		BLCK0018
1	28,010, 44,010, 28,016, 39,944, 30,008, 14,008,		BLCK0019
1	33,008, 46,008, 17,032, 1,000,		BLCK0020
4	59*0./		BLCK0021
C			BLCK0022
	END		BLCK0023
			BLCK0024

CBLKRDA	-BLKRDA-	BLOCK DATA	(REACTION DATA)	BLKRDA01
C				BLKRDA02
		BLOCK DATA		BLKRDA03
C				BLKRDA04
*		REACTION DATA (LSR = A - N - EACT - M - NEGL)		BLKRDA05
C				BLKRDA06
		COMMON / LSLRSV / LS, LR		BLKRDA07
C				BLKRDA08
		COMMON / RFAC / LSR(4,30), DUMREA(15)		BLKRDA09
C				BLKRDA10
		COMMON / RRAT / A(30), N(30), EACT(30), RDUM(30), M(25,30), DUMAL		BLKRDA11
C				BLKRDA12
		REAL M, N		BLKRDA13
C				BLKRDA14
		COMMON / SKIP / NEGL(25), I1, I2, IT		BLKRDA15
C				BLKRDA16
		DATA LS, LR/16, 23/		BLKRDA17
C				BLKRDA18
		DATA I1, IT/7, 7/		BLKRDA19
C				BLKRDA20
		DATA (NEGL(I), I=1, 7)/		BLKRDA21
		* 1, 2, 10, 12, 13, 15, 16/		BLKRDA22
C				BLKRDA23
		DATA ((LSR(I, J), I=1, 4), J=1, 19)/		BLKRDA24
		1 7, 5, 8, 1,		BLKRDA25
		2 7, 4, 8, 2,		BLKRDA26
		3 1, 5, 6, 0,		BLKRDA27
		4 0, 3, 1, 1,		BLKRDA28
		5 1, 2, 5, 0,		BLKRDA29
		6 2, 2, 4, 0,		BLKRDA30
		7 5, 3, 6, 1,		BLKRDA31
		8 1, 4, 5, 2,		BLKRDA32
		9 2, 3, 5, 1,		BLKRDA33
		J 6, 2, 5, 5,		BLKRDA34
		K 3, 4, 5, 5,		BLKRDA35
		L 12, 11, 9, 2,		BLKRDA36
		M 12, 4, 11, 2,		BLKRDA37
		N 12, 5, 11, 1,		BLKRDA38
		O 1, 4, 13, 0,		BLKRDA39
		P 1, 13, 5, 5,		BLKRDA40
		Q 5, 13, 6, 4,		BLKRDA41
		R 2, 13, 5, 4,		BLKRDA42
		S 1, 13, 3, 4/		BLKRDA43
C				BLKRDA44
		DATA ((LSR(I, J), I=1, 4), J=20, 23)/		BLKRDA45
		T 11, 13, 14, 5,		BLKRDA46
		U 14, 1, 11, 5,		BLKRDA47
		V 14, 2, 11, 4,		BLKRDA48
		W 11, 2, 14, 0/		BLKRDA49
C				BLKRDA50

DATA A/
 * 1.1206042E+3, 2.50000E+12, 1.00000E+19, 1.120000E+13,
 * 5.30000E+15, 8.15000E+18, 2.30000E+13, 2.00000E+14,
 * 4.00000E+13, 8.40000E+13, 8.00000E+14, 3.10000E+13,
 * 6.40000E+09, 4.00000E+13, 1.00000E+15, 7.00000E+13,
 * 6.00000E+12, 6.00000E+12, 2.30000E+13, 1.00000E+13,
 * 7.20000E+14, 1.90000E+13, 9.40000E+14,
 * 7*0./

BLKRDA51
 BLKRDA52
 BLKRDA53
 BLKRDA54
 BLKRDA55
 BLKRDA56
 BLKRDA57
 BLKRDA58
 BLKRDA59

DATA N/
 * 2.4863852, 0., -1.0000, 0.5000, 0., -1.2200,
 * 6*0., 1.0, 17*0./

BLKRDA60
 BLKRDA61
 BLKRDA62
 BLKRDA63
 BLKRDA64
 BLKRDA65
 BLKRDA66
 BLKRDA67
 BLKRDA68

DATA EACT/
 * -2574.7429, 40000.00, 0., 92600.00, -2780.00, 0.,
 * 5200.00, 16500.00, 10200.00, 18000.00, 45000.00, 334.00,
 * 6250.00, 0., -1300.00, 5*0., 1930.00, 1060.00, -1930.00,
 * 7*0./

BLKRDA69
 BLKRDA70
 BLKRDA71
 BLKRDA72
 BLKRDA73
 BLKRDA74
 BLKRDA75
 BLKRDA76
 BLKRDA77

DATA ((M(I,J), I=1,16), J=1,9) /
 1 1. 1. 1. 1. 1. 1. 1. 1. 1.
 * 1. 1. 1. 1. 1. 1. 1. 1. 1.
 2 1. 1. 1. 1. 1. 1. 1. 1. 1.
 * 1. 1. 1. 1. 1. 1. 1. 1. 1.
 3 1.07, 1.07, 4.5, 4.5, 4.5, 18.0, 4.5, 8.0,
 * 3.6, 1.0, 4.5, 1. 1. 1. 1. 1. 1.
 4 1. 1. 1. 1. 1. 1. 1. 1. 1.
 * 1. 1. 1. 1. 1. 1. 1. 1. 1.
 5 1. 1. 1. 1. 1. 1. 1. 1. 1.
 * 1. 1. 1. 1. 1. 1. 1. 1. 1.
 6 1. 1. 1. 1. 1. 1. 1. 1. 1.
 * 1. 1. 1. 1. 1. 1. 1. 1. 1.
 7 1. 1. 1. 1. 1. 1. 1. 1. 1.
 * 1. 1. 1. 1. 1. 1. 1. 1. 1.
 8 1. 1. 1. 1. 1. 1. 1. 1. 1.
 * 1. 1. 1. 1. 1. 1. 1. 1. 1.
 9 1. 1. 1. 1. 1. 1. 1. 1. 1.
 * 1. 1. 1. 1. 1. 1. 1. 1. 1.

BLKRDA78
 BLKRDA79
 BLKRDA80
 BLKRDA81
 BLKRDA82
 BLKRDA83
 BLKRDA84
 BLKRDA85
 BLKRDA86
 BLKRDA87

DATA ((M(I,J), I=1,16), J=10,16) /
 1 1. 1. 1. 1. 1. 1. 1. 1. 1.
 * 1. 1. 1. 1. 1. 1. 1. 1. 1.
 1 1. 1. 1. 1. 1. 1. 1. 1. 1.
 1 1. 1. 1. 1. 1. 1. 1. 1. 1.
 1 1. 1. 1. 1. 1. 1. 1. 1. 1.
 2 1. 1. 1. 1. 1. 1. 1. 1. 1.
 1 1. 1. 1. 1. 1. 1. 1. 1. 1.
 3 1. 1. 1. 1. 1. 1. 1. 1. 1.
 1 1. 1. 1. 1. 1. 1. 1. 1. 1.
 4 1. 1. 1. 1. 1. 1. 1. 1. 1.
 1 1. 1. 5.0, 2.0, 2.0, 32.5, 2.0, 7.5,

BLKRDA88
 BLKRDA89
 BLKRDA90
 BLKRDA91
 BLKRDA92
 BLKRDA93
 BLKRDA94
 BLKRDA95
 BLKRDA96
 BLKRDA97
 BLKRDA98
 BLKRDA99
 BLKRDA00

CBLKST	STOICHIOMETRIC COEFFICIENTS	
	BLOCK DATA	BLKST001
C		BLKST002
		BLKST003
CA	STOICHIOMETRIC COEFFICIENTS FOR REACTIONS	BLKST004
C		BLKST005
	COMMON / SPEC / DUM1(110), STOIC(25,30), DUM2(750)	BLKST006
C		BLKST007
	DATA ((STOIC(I,J), I=1,16), J=1,9) /	BLKST008
	1 1., 0., 0., 0., -1., 0., -1., 1.,	BLKST009
	*0., 0., 0., 0., 0., 0., 0., 0.,	BLKST010
	2 0., 1., 0., -1., 0., 0., -1., 1.,	BLKST011
	*0., 0., 0., 0., 0., 0., 0., 0.,	BLKST012
	3 -1., 0., 0., 0., -1., 1., 0., 0.,	BLKST013
	*0., 0., 0., 0., 0., 0., 0., 0.,	BLKST014
	4 2., 0., -1., 0., 0., 0., 0., 0.,	BLKST015
	*0., 0., 0., 0., 0., 0., 0., 0.,	BLKST016
	5 -1., -1., 0., 0., 1., 0., 0., 0.,	BLKST017
	*0., 0., 0., 0., 0., 0., 0., 0.,	BLKST018
	6 0., -2., 0., 1., 0., 0., 0., 0.,	BLKST019
	*0., 0., 0., 0., 0., 0., 0., 0.,	BLKST020
	7 1., 0., -1., 0., -1., 1., 0., 0.,	BLKST021
	*0., 0., 0., 0., 0., 0., 0., 0.,	BLKST022
	8 -1., 1., 0., -1., 1., 0., 0., 0.,	BLKST023
	*0., 0., 0., 0., 0., 0., 0., 0.,	BLKST024
	9 1., -1., -1., 0., 1., 0., 0., 0.,	BLKST025
	*0., 0., 0., 0., 0., 0., 0., 0. /	BLKST026
C		BLKST027
	DATA ((STOIC(I,J), I=1,16), J=10,18) /	BLKST028
	1 0., -1., 0., 0., 2., -1., 0., 0.,	BLKST029
	*0., 0., 0., 0., 0., 0., 0., 0.,	BLKST030
	1 0., 0., -1., -1., 2., 0., 0., 0.,	BLKST031
	10., 0., 0., 0., 0., 0., 0., 0.,	BLKST032
	1 0., 1., 0., 0., 0., 0., 0., 0.,	BLKST033
	2 1., 0., -1., -1., 0., 0., 0., 0.,	BLKST034
	1 0., 1., 0., -1., 0., 0., 0., 0.,	BLKST035
	3 0., 0., 1., -1., 0., 0., 0., 0.,	BLKST036
	1 1., 0., 0., 0., -1., 0., 0., 0.,	BLKST037
	4 0., 0., 1., -1., 0., 0., 0., 0.,	BLKST038
	1 -1., 0., 0., -1., 0., 0., 0., 0.,	BLKST039
	5 0., 0., 0., 0., 1., 0., 0., 0.,	BLKST040
	1 -1., 0., 0., 0., 2., 0., 0., 0.,	BLKST041
	6 0., 0., 0., 0., -1., 0., 0., 0.,	BLKST042
	1 0., 0., 0., 1., -1., 1., 0., 0.,	BLKST043
	7 0., 0., 0., 0., -1., 0., 0., 0.,	BLKST044
	1 0., -1., 0., 1., 1., 0., 0., 0.,	BLKST045
	8 0., 0., 0., 0., -1., 0., 0., 0. /	BLKST046
C		BLKST047
	DATA ((STOIC(I,J), I=1,16), J=19,23) /	BLKST048
	1 -1., 0., 1., 1., 0., 0., 0., 0.,	BLKST049
	9 0., 0., 0., 0., -1., 0., 0., 0.,	BLKST050

```

2 0., 0., 0., 0., 1., 0., 0., 0.,
* 0., 0., -1., 0., -1., 1., 0., 0.,
2 -1., 0., 0., 0., 1., 0., 0., 0.,
1 0., 0., 1., 0., 0., -1., 0., 0.,
2 0., -1., 0., 1., 0., 0., 0., 0.,
2 0., 0., 1., 0., 0., -1., 0., 0.,
2 0., -1., 0., 0., 0., 0., 0., 0.,
3 0., 0., -1., 0., 0., 1., 0., 0.,

```

C

END

```

BLKST051
BLKST052
BLKST053
BLKST054
BLKST055
BLKST056
BLKST057
BLKST058
BLKST059
BLKST060

```

BLKTH	BLKTH=	BLOCK DATA (THERMO)	BLKTH001
C			BLKTH002
	BLOCK DATA		BLKTH003
C			BLKTH004
*	THERMODYNAMIC DATA		BLKTH005
C			BLKTH006
C			BLKTH007
	COMMON / TCOF / TC(7,2,25), TLOW, TMID, THI		BLKTH008
	COMMON / COND / DUMC1(33), LS, DUMC2(2)		BLKTH009
C			BLKTH010
	DATA LS/16/		BLKTH011
	DATA TLOW, TMID, THI/300.0,1000.0,5000.0/		BLKTH012
C			BLKTH013
C			BLKTH014
	DATA (((TC(K,KK,I),K=1,7),KK=1,2),I=1,3)/		BLKTH015
*	0.25000000E 01, 0. , 0. , 0.25981225E 05,		BLKTH016
*	0. , 0. , 0. , 0.25981225E 05,		BLKTH017
*	-0.46011763E 00, 0.25000000E 01, 0. ,		BLKTH018
*	0. , 0. , 0. , 0. ,		BLKTH019
*	0.25981225E 05, -0.46011762E 00,		BLKTH020
*	0.25420596E 01, -0.27550619E-04, -0.31028033E-08,		BLKTH021
*	0.45510674E-11, -0.43680515E-15, 0.29753241E 05,		BLKTH022
*	0.49203080E 01, 0.29464287E 01, -0.16381665E-02,		BLKTH023
*	0.24210316E-05, -0.16028432E-08, 0.38906964E-12,		BLKTH024
*	0.29670083E 05, 0.29639949E 01,		BLKTH025
*	0.31001901E 01, 0.51119464E-03, 0.52644210E-07,		BLKTH026
*	-0.34909973E-10, 0.36945345E-14, 0.14181509E 03,		BLKTH027
*	-0.19629421E 01, 0.30574451E 01, 0.26765200E-02,		BLKTH028
*	-0.58099162E-05, 0.55210391E-08, -0.18122739E-11,		BLKTH029
*	0.30290764E 02, -0.22997056E 01/		BLKTH030
C			BLKTH031
	DATA (((TC(K,KK,I),K=1,7),KK=1,2),I=4,6)/		BLKTH032
*	0.36219535E 01, 0.73618264E-03, -0.19652228E-06,		BLKTH033
*	0.36201558E-10, -0.28945627E-14, -0.15710570E 03,		BLKTH034
*	0.36150960E 01, 0.36255985E 01, -0.18782184E-02,		BLKTH035
*	0.70554544E-05, -0.67635137E-08, 0.21555993E-11,		BLKTH036
*	-0.26457977E 01, 0.43052778E 01,		BLKTH037
*	0.29131230E 01, 0.95418248E-03, -0.19084325E-06,		BLKTH038
*	0.12730795E-10, 0.24803941E-15, 0.49967421E 04,		BLKTH039
*	0.54288735E 01, 0.38365518E 01, -0.10702014E-02,		BLKTH040
*	0.94849757E-06, 0.20843575E-09, -0.23384265E-12,		BLKTH041
*	0.47036168E 04, 0.49805456E 00,		BLKTH042
*	0.27167633E 01, 0.29451374E-02, -0.80224374E-06,		BLKTH043
*	0.10226682E-09, -0.48472145E-14, -0.28364192E 05,		BLKTH044
*	0.66305671E 01, 0.40701275E 01, -0.11084499E-02,		BLKTH045
*	0.41521180E-05, -0.29637404E-08, 0.80702103E-12,		BLKTH046
*	-0.28738088E 05, 0.32270046E 00/		BLKTH047
C			BLKTH048
	DATA (((TC(K,KK,I),K=1,7),KK=1,2),I=7,9)/		BLKTH049
*	0.29840696E 01, 0.14891390E-02, -0.57899684E-06,		BLKTH050

* 0.10364577E-09, -0.69353550E-14, -0.13595894E-05,	BLKTH051
* 0.63479156E-01, 0.37100928E-01, -0.16190964E-02,	BLKTH052
* 0.36923594E-05, -0.20319674E-08, 0.23953384E-12,	BLKTH053
* -0.13706976E-05, 0.29555351E-01,	BLKTH054
* 0.44608041E-01, 0.30981719E-02, -0.12392571E-05,	BLKTH055
* 0.22741325E-09, -0.15525954E-13, -0.47789669E-05,	BLKTH056
* -0.98635982E+00, 0.24007797E+01, 0.87350957E-02,	BLKTH057
* -0.66070878E-05, 0.20021861E-08, 0.63274039E-15,	BLKTH058
* -0.47205754E-05, 0.96951457E-01,	BLKTH059
* 0.28963194E-01, 0.15154866E-02, -0.57235277E-06,	BLKTH060
* 0.99807393E-10, -0.65223555E-14, 0.13750436E-03,	BLKTH061
* 0.61815148E-01, 0.36748261E-01, -0.12081500E-02,	BLKTH062
* 0.23240102E-05, -0.63217559E-09, -0.22577253E-12,	BLKTH063
* -0.17792603E-02, 0.23580424E-01,	BLKTH064

C

DATA (((TC(K,KK,I),K=1,7),KK=1,2),I=10,12)/	
* 0.25000000E-01, 0.	BLKTH066
* 0.	BLKTH067
* 0.43660006E-01, 0.25000000E-01, 0.	BLKTH068
* 0.	BLKTH069
* 0.45776367E-04, 0.43660006E+01,	BLKTH070
* 0.31890007E-01, 0.13382281E-02, -0.52899318E-06,	BLKTH071
* 0.95919332E-10, -0.64847932E-14, 0.10872450E-05,	BLKTH072
* 0.67458126E-01, 0.40459521E-01, -0.34181783E-02,	BLKTH073
* 0.79819190E-05, -0.61139316E-08, 0.15919076E-11,	BLKTH074
* 0.10789515E-05, 0.29974988E-01,	BLKTH075
* 0.24502682E-01, 0.10661458E-03, -0.74653373E-07,	BLKTH076
* 0.18796524E-10, -0.10259839E-14, 0.56637723E-05,	BLKTH077
* 0.44487581E-01, 0.25030714E-01, -0.21800181E-04,	BLKTH078
* 0.54205287E-07, -0.56475602E-10, 0.20999044E-13,	BLKTH079
* 0.56620587E-05, 0.41675764E-01,	BLKTH080

C

DATA (((TC(K,KK,I),K=1,7),KK=1,2),I=13,15)/	
* 0.37866280E-01, 0.27885404E-02, -0.10168708E-05,	BLKTH081
* 0.17183946E-09, -0.11021852E-13, 0.27433246E-04,	BLKTH082
* 0.48147611E-01, 0.35094850E-01, 0.11499670E-02,	BLKTH083
* 0.58784259E-05, -0.77795519E-08, 0.29607883E-11,	BLKTH084
* 0.29348076E-04, 0.68276325E-01,	BLKTH085
* 0.46240771E-01, 0.25260332E-02, -0.10609498E-05,	BLKTH086
* 0.19879239E-09, -0.13799384E-13, 0.38565499E-04,	BLKTH087
* 0.13324138E-01, 0.34584236E-01, 0.20647064E-02,	BLKTH088
* 0.66866067E-05, -0.95556725E-08, 0.36195881E-11,	BLKTH089
* 0.43817864E-04, 0.83116983E-01,	BLKTH090
* 0.24165177E-01, 0.61871211E-02, -0.21785136E-05,	BLKTH091
* 0.37599090E-09, -0.24448856E-13, -0.44242413E-04,	BLKTH092
* 0.77043482E-01, 0.35912768E-01, 0.49388668E-03,	BLKTH093
* 0.83449322E-05, -0.83833385E-08, 0.27299092E-11,	BLKTH094
* -0.46212379E-04, 0.22520966E-01,	BLKTH095

C

DATA (((TC(K,KK,I),K=1,7),KK=1,2),I=16,16)/	
	BLKTH096
	BLKTH097
	BLKTH098
	BLKTH099
	BLKTH100

1 3.4557152,1.1491803E-2,-4.3651750E-6,
6 7.6155095E-10,-5.0123200E-14,6.7694946E3,2.6987959,
* 1.4256821.1.1383140E-2,7.9890006E-6,-1.6253679E-8,
* 6.7491256E-12,7.6292582E3,1.4621819E1/

C

END

BLKTH101
BLKTH102
BLKTH103
BLKTH104
BLKTH105
BLKTH106

*BLOCK1	BLOCK DATA FOR TURBULENT PARAMETERS	BLOCKT01
	BLOCK DATA	BLOCKT02
	REAL MUL,MUEFF,KCP,MUREF,MACH	BLOCKT03
	COMMON /CPRDP/ CT(10)	BLOCKT04
	COMMON /CPRDP2/ CTP , CT8 , CTM	BLOCKT05
C*		BLOCKT06
	COMMON /PRNPJT/	BLOCKT07
	* P , PRL , PRT , RGAS , SC ,	BLOCKT08
	* TREF , MUREF , MACH , XLC ,	BLOCKT09
	* REFI , C , CHI , RNDRM ,	BLOCKT10
	* RHJ(200) , MUL(200) , KCP(200) ,	BLOCKT11
	* MUEFF(200) , XLN(200) , QK(200) , RETURB(200)	BLOCKT12
C*		BLOCKT13
C*		BLOCKT14
	DATA	BLOCKT15
	* C/2.59/ , PRL/.72/ , PRT/1./ , RGAS/53.34/ ,	BLOCKT16
	* TREF/0./ , MUREF/0./ , SC/216./ ,	BLOCKT17
	* CHI/.586/ , RNDRM/110./	BLOCKT18
	DATA CT/.23/.38/.23/.38/1.4/.43/.1875/.9.0./	BLOCKT19
	DATA CTP/.175/ , CTM/.23/ , CT8/.23/	BLOCKT20
C*		BLOCKT21
	END	BLOCKT22

CCARDL	FUEL-AIR RATIO AND EMISSION INDICES FROM CAROL ANALYSIS	CARDL001
C	OF GAS SAMPLE	CARDL002
	SUBROUTINE CAROL(N,WAR,RHC,RCO,RCO2,RNOX,FAR,EIHC,FICO,EINOX)	CARDL003
	REAL N,I,MD,MT,MWTF,MWT	CARDL004
	XM(X,Y)=(X*(F-C)-Y*(I-C))/((G-A)*(E-B)-(D-A)*(H-B))	CARDL005
C		CARDL006
	A=1.0-(0.1-0.25*N)*RHC	CARDL007
	B=-0.503101*(RCO+RH2(RCO))	CARDL008
	C=-(0.03452247+WAR/18.016)	CARDL009
	D=(0.1+0.25*N)*RHC	CARDL010
	E=0.503101*(RCO+3.0*RH2(RCO))-1.0	CARDL011
	F=0.03452247	CARDL012
	G=0.25*N*RHC	CARDL013
	H=0.2515505*N*(RCO+RCO2)	CARDL014
	I=-2.5891851E-6*N	CARDL015
C		CARDL016
	MD=-XM(G-A,D-A)	CARDL017
	MT=XM(H-B,F-B)	CARDL018
	MWTF=120.1+10.08*N	CARDL019
	FAR=0.4*MWTF/N*(A*MT+B*MD+C)	CARDL020
	GPF=(1.0+FAR+WAR)/FAR	CARDL021
	MWT=GPF*FAR/MT	CARDL022
C		CARDL023
	EIHC=RHC/10.0*MWTF/MWT*GPF	CARDL024
	FICO=RCO*1.006202*MD/MT*28.01/MWT*GPF	CARDL025
	EINOX=RNOX*46.008/MWT*GPF	CARDL026
C		CARDL027
	RETURN	CARDL028
	END	CARDL029

CCASMM	CHOOSE STEP FORMULA, SET UP AUGMENTED MATRIX, INCREMENTS	CASMM001
	SUBROUTINE CASM	CASMM002
C		CASMM003
C	CHOOSE (1) INITIAL STEP (RESTART) FORMULA	CASMM004
C	(2) GENERAL STEP FORMULA	CASMM005
C		CASMM006
C	SET UP AUGMENTED MATRIX	CASMM007
C		CASMM008
C	COMPUTE INCREMENTS	CASMM009
C		CASMM010
	DOUBLE PRECISION A	CASMM011
C		CASMM012
	REAL IVAR	CASMM013
C		CASMM014
	COMMON/DPTS/VERST, TIMEV, VERSA, AREAV, FLIM, ICN, RHOCON, IPRCOD	CASMM015
	COMMON/C7ND/DVAR, AREA, MDO1, P, IVAR, Y(28), LS, LSP3, NEXT	CASMM016
	COMMON/CFRM/F(28), ALPHA(28), BETA(28,28)	CASMM017
	COMMON/MATX/A(28,29)	CASMM018
	DIMENSION QK(28), RK(28)	CASMM019
	EQUIVALENC (VN1,Y(1))	CASMM020
C		CASMM021
	DATA EPS/0.0001/	CASMM022
C		CASMM023
	ENTRY CASI (HN,QK,HN1,RK)	CASMM024
C		CASMM025
C		CASMM026
C	INITIAL STEP OR RESTART	CASMM027
	INGEN = 1	CASMM028
	JK = 0	CASMM029
	NEGS=0	CASMM030
10	F1 = 0.	CASMM031
	F2 = HN1	CASMM032
	F3 = HN1/2	CASMM033
	F4 = F3	CASMM034
	GO TO 2	CASMM035
C		CASMM036
	ENTRY CASG (HN,QK,HN1,RK)	CASMM037
C	GENERAL STEP	CASMM038
	INGEN = 2	CASMM039
	JK = 0	CASMM040
	NEGS=0	CASMM041
20	F1 = HN1*HN1/((2.*HN1 + HN)*HN)	CASMM042
	F2 = HN1*(HN1 + HN)/(2.*HN1 + HN)	CASMM043
	F3 = HN1	CASMM044
	F4 = F2	CASMM045
C		CASMM046
	2 LSP4 = LSP3 + 1	CASMM047
C		CASMM048
	DO 4 I=1,LSP3	CASMM049
	DO 3 J=1,LSP3	CASMM050

3	A(I,J) = -F4*BETA(I,J)	CASMM051
	A(I,I) = 1.00 + A(I,I)	CASMM052
	IF (DARS(A(I,I)) .GE. EPS) GO TO 4	CASMM053
	IF (JK .GE. 3) GO TO 4	CASMM054
	JK = JK + 1	CASMM055
	HN1 = 2.*((1. - FPS)/BETA(I,I))	CASMM056
	GO TO (10,20),INGEN	CASMM057
4	A(I,LSP4) = F1*QK(I) + F2*(F(I) + ALPHA(I)*F3)	CASMM058
C		CASMM059
	CALL LESV (RK)	CASMM060
	DO 7 I=1,LSP3	CASMM061
	IF (Y(I)+RK(I) .GE. 0.0) GO TO 7	CASMM062
	NEGS = NEGS+1	CASMM063
	IF (NEGS .GT. 4) GO TO 7	CASMM064
	HN1 = 0.5*HN1	CASMM065
	GO TO (10,20),INGEN	CASMM066
7	CONTINUE	CASMM067
	VN = Y(1)	CASMM068
	DO 5 I=1,LSP3	CASMM069
5	Y(I) = Y(I) + RK(I)	CASMM070
	IVAR = IVAR + HN1	CASMM071
C		CASMM072
	IF (VERSI .EQ. TIMEV) GO TO 6	CASMM073
	DVAR = DVAR + 2.*HN1/(VN + VN1)	CASMM074
	RETURN	CASMM075
6	DVAR = DVAR + (VN + VN1)/2.*HN1	CASMM076
	RETURN	CASMM077
C		CASMM078
	END	CASMM079

CCFILE MERGE OR SPLIT FILES
SUBROUTINE MAIN

C

COMMON /FILK / CSC

COMMON /CPRJET/ NPIS,RAD(12),TS(12),UJ(12),SPV(12),
* MWT(12),CP(12),FUEL(12),SPALDG(12),TKE(12),OTHER(36),
* TITLE(20),PRINT(20),GEX,N,HE,WAR,T2,BETA,T25,FARS,EIND2C,
* P0,RCU(11),RCU2(11),RHC(11),RNJX(11),PT(11),PS(11),
* BLOC(11),GCC(11),RICH(24),FUJL(48),ENTH(48),CONC(768),
* HCINCP(24),OTHER2(192)
COMMON /CGJET / GJET(200)

C

C

COMMON /CJMIX1/ NAME(10),TITLE1(10),IDENT(10),ADDRES(10),
* IDENT1(10),INOUT(9),BITS,ERK,GC,GCC,FOOT,DIAJ,MJET,TJFT,
* PTJET,VJET,TJET,EJET,PE,YE,ME,TIF,TE,AXI,NJ,NM,UE,
* MIXPRE,XLC,FLOWJ,MERGE,NV,CONV,CT1,CT2,CT3,CT4,CT5,CT6,
* CT7,CT8,CT9,CTP,CTS,CTM,GAM,RG,PR,PRT,SC,TREF,MUREF,
* SP,SV,SLEN,DPRIN,PLDT,C6,MIX,CF,MAXIT,TOL,SUPB,
* X(100),XPRN(100),B(100),UC(100),TC(100),TIC(100),
* PTC(100),WJ(100),YJ(100),TTC(100),YSONIC(100),YCB(100),
* XD(100),RD(100),YR(100),YCD(100),PD(100),WV(100),
* MA2(100),VEP(100),TE2(100),NXTA,I,NC,CNAME(12),ALJ(12),
* ALJD(12),ALE(12),SCM(12),CPC(36),DIFF

C

COMMON /CJMIX2/ SUPD,SUPSTP,CORE,CORSTP,MER,MERSTP,NPD,
* PSI(200),Y(200),JD(200),ED(200),TJD(200),RHO(200),
* XLN(200),U(200),IDT(200),XMACH(200),PIOT(200),ITD(200),
* PTD(200),MOLF1(100),MOLF2(100),MOLF3(100),MOLF4(100),
* MOLF5(100),MOLF6(100),MOLF7(100),MOLF8(100),MOLF9(100),
* MOLF10(100),MOLF11(100),MOLF12(100),T(200)

C

COMMON /CSPALD/ SY(50),RADS(50),GS(50),JEND

C

COMMON /TROUHL/ ERR,ERRMAJ,INERR,PRERR
LOGICAL ERR,ERRMAJ,INERR,PRERR
LOGICAL MERGE , SPLIT
NAMELIST /A/ MERGE,SPLIT,NF

C

MERGE = 1 MERGE FILES
SPLIT = 1 SPLIT OFF FILES
NF FILE PRESENT
= 1 PREJET
= 2 PREJET,JETMIX
= 3 PREJET,JETMIX,SPALDG

C

CALL FLGERR(4,ERR)
I MERGE = .FALSE.
SPLIT = .FALSE.
NF = 0

CFILE001
CFILE002
CFILE003
CFILE004
CFILE005
CFILE006
CFILE007
CFILE008
CFILE009
CFILE010
CFILE011
CFILE012
CFILE013
CFILE014
CFILE015
CFILE016
CFILE017
CFILE018
CFILE019
CFILE020
CFILE021
CFILE022
CFILE023
CFILE024
CFILE025
CFILE026
CFILE027
CFILE028
CFILE029
CFILE030
CFILE031
CFILE032
CFILE033
CFILE034
CFILE035
CFILE036
CFILE037
CFILE038
CFILE039
CFILE040
CFILE041
CFILE042
CFILE043
CFILE044
CFILE045
CFILE046
CFILE047
CFILE048
CFILE049
CFILE050

READ (5,A)	CFILE051
IF(ERR) RETURN	CFILE052
IF(SPLIT) GO TO 500	CFILE053
C	CFILE054
5 WRITE (6,7)	CFILE055
7 FORMAT(1H1,10X,17H** FILE MERGE **/10X,14HDATA ON FILE 4//	CFILE056
* 10X,6HPREJET)	CFILE057
C	CFILE058
10 WRITE (4) NF	CFILE059
C	CFILE060
C	CFILE061
C*	CFILE062
SECTION TO MERGE FILES (JOB TERMINATION)	CFILE063
C	CFILE064
C*	CFILE065
PREJET **	CFILE066
C	CFILE067
12 READ (1)	CFILE068
* DIAJ,MJET,TIJET,VJET,PE,TF,TIE,VE,GFX,RG,PR,PRT,SC,TREF,	CFILE069
* MUREF,DIFF,NC,CNAME,ALF,SCM,CPC,NJ,NM,CT1,CT2,CT3,CT4,CT5,	CFILE070
* CT6,CT7,CT8,GJET,Y,UD,THD,TID,ED,	CFILE071
* MOLF1,MOLF2,MOLF3,MOLF4,MOLF5,MOLF6,	CFILE072
* MOLF7,MOLF8,MOLF9,MOLF10,MOLF11,MOLF12,	CFILE073
* NPTS,RAD,IS,UJ,SPV,MAT,CP,FUEL,SPALDG,TKE,OTHER,	CFILE074
* TITLE,PRINT,N,HF,WAR,T2,BETA,T25,FARS,EIND2C,P0,	CFILE075
* RCD,RCD2,RMC,RNOX,PT,PS,BLOC,QCD,	CFILE076
* RICH,HCINCP,FUUL,ENTH,CONC,OTHER2	CFILE077
* ,GEX	CFILE078
14 WRITE (4)	CFILE079
* DIAJ,MJET,TIJET,VJET,PE,TF,TIE,VE,GFX,RG,PR,PRT,SC,TREF,	CFILE080
* MUREF,DIFF,NC,CNAME,ALF,SCM,CPC,NJ,NM,CT1,CT2,CT3,CT4,CT5,	CFILE081
* CT6,CT7,CT8,GJET,Y,UD,THD,TID,ED,	CFILE082
* MOLF1,MOLF2,MOLF3,MOLF4,MOLF5,MOLF6,	CFILE083
* MOLF7,MOLF8,MOLF9,MOLF10,MOLF11,MOLF12,	CFILE084
* NPTS,RAD,TS,UJ,SPV,MAT,CP,FUEL,SPALDG,TKE,OTHER,	CFILE085
* TITLE,PRINT,N,HF,WAR,T2,BETA,T25,FARS,EIND2C,P0,	CFILE086
* RCD,RCD2,RMC,RNOX,PT,PS,BLOC,QCD,	CFILE087
* RICH,HCINCP,FUUL,ENTH,CONC,OTHER2	CFILE088
* ,GEX	CFILE089
IF(NF.EQ.1) GO TO 1000	CFILE090
C	CFILE091
C*	CFILE092
JEIMIX **	CFILE093
C	CFILE094
20 WRITE (6,21)	CFILE095
21 FORMAT(10X,6HJETMIX,18X,2HXX)	CFILE096
23 READ (2) KXX1,KREC,	CFILE097
* NAME,TITLE1,IDENT,ADDRES,IDENT1,TWONT,BJTS,FRK,Gc,GcJ,	CFILE098
* FOOT,DIAJ,MJET,TJET,PTJET,VJET,TIJET,EJET,PE,VE,ME,T1,TE,	CFILE099
* AXI,NJ,NM,UE,MIXPRF,XLC,FIOWJ,MERGE,NV,CON1,	CFILE100
* CT1,CT2,CT3,CT4,CT5,CT6,CT7,CT8,CT9,CTP,CTS,CTM,	
* GAM,RG,PR,PRT,SC,TREF,MUREF,SP,SV,SLEN,DPRIN,PLOT,C6,	
* MIX,CF,MAXIT,TOL,SUPH,	

* X,XPRN,B,UC,TC,TIC,PTC,WJ,YJ,TTC,YSONIC,	CFILE101
* YCB,XD,RD,YR,YCD,PD,WV,MA2,VE2,TE2,NXTA,I,	CFILE102
* NC,CNAME,ALJ,ALJO,ALE,SCM,CPC,DIFF,CSC	CFILE103
WRITE (4) KXX1,KREC,	CFILE104
* NAME,TITLE1,IDENT,ADDRES,IDENT1,TWONT,BITS,ERK,GC,GCJ,	CFILE105
* FOJT,DIAJ,MJET,TJET,PTJET,VJET,TIJET,EJET,PE,VE,ME,TIE,TE,	CFILE106
* AXI,NJ,NM,UE,MIXPRE,XLC,FLOWJ,MERGE,NV,CON1,	CFILE107
* CT1,CT2,CT3,CT4,CT5,CT6,CT7,CT8,CT9,CTP,CTS,CTM,	CFILE108
* GAM,RG,PR,PRT,SC,TREF,MUREF,SP,SV,SI,EN,DPRIN,PLOT,C6,	CFILE109
* MIX,CE,MAX11,IDL,SUPB,	CFILE110
* X,XPRN,B,UC,TC,TIC,PTC,WJ,YJ,TTC,YSONIC,	CFILE111
* YCB,XD,RD,YR,YCD,PD,WV,MA2,VE2,TE2,NXTA,I,	CFILE112
* NC,CNAME,ALJ,ALJO,ALE,SCM,CPC,DIFF,CSC	CFILE113
DO 25 IJ=1,KREC	CFILE114
READ (2) JRFC,KXX,KREG,	CFILE115
* SUPD,SUPSTP,CORE,CORSTP,NER,MERSIP,NPD,	CFILE116
* PSI,Y,UD,THD,ED,TID,RHO,XLN,U,T,TOT,XMACH,	CFILE117
* PTOT,TID,PID,	CFILE118
* MOLF1,MOLF2,MOLF3,MOLF4,MOLF5,MOLF6,	CFILE119
* MOLF7,MOLF8,MOLF9,MOLF10,MOLF11,MOLF12,J	CFILE120
WRITE (4) JRFC,KXX,KREG,	CFILE121
* SUPD,SUPSTP,CORE,CORSTP,NER,MERSIP,NPD,	CFILE122
* PSI,Y,UD,THD,ED,TID,RHO,XLN,U,T,TOT,XMACH,	CFILE123
* PTOT,TID,PID,	CFILE124
* MOLF1,MOLF2,MOLF3,MOLF4,MOLF5,MOLF6,	CFILE125
* MOLF7,MOLF8,MOLF9,MOLF10,MOLF11,MOLF12,J	CFILE126
XKXX = FLOAT(KXX)/CSC	CFILE127
WRITE (6,24) XKXX	CFILE128
24 FORMAT(10X,6HJETMIX,F20.3)	CFILE129
25 CONTINUE	CFILE130
C	CFILE131
IF(NF.EQ.2) GO TO 1000	CFILE132
C	CFILE133
C* SPALDG **	CFILE134
C	CFILE135
30 DO 35 IJ=1,KREC	CFILE136
READ (3) KXX,SY,RADS,GS,JEND	CFILE137
WRITE (4) KXX,SY,RADS,GS,JEND	CFILE138
XKXX = FLOAT(KXX)/CSC	CFILE139
WRITE (6,34) XKXX	CFILE140
34 FORMAT(10X,6HSPALDG,F20.3)	CFILE141
35 CONTINUE	CFILE142
GO TO 1000	CFILE143
C	CFILE144
C* SECTION TO SPLIT FILES (JOB INITIATION)	CFILE145
C	CFILE146
500 READ (7) NF	CFILE147
WRITE (5,605)	CFILE148
READ (7)	CFILE149
* DIAJ,MJET,TJET,VJET,PE,TE,TIE,VE,GEX,RG,PR,PRT,SC,TREF,	CFILE150

* MURFF,DIFF,NC,CNAME,ALE,SCM,CPC,NJ,NM,CT1,CT2,CT3,CT4,CT5,	CFILE151
* CT6,CT7,CT8,GJET,Y,UD,THD,TID,ED,	CFILE152
* MOLF1,MOLF2,MOLF3,MOLF4,MOLF5,MOLF6,	CFILE153
* MOLF7,MOLF8,MOLF9,MOLF10,MOLF11,MOLF12,	CFILE154
* NPIS,RAD,TS,UJ,SPV,MWT,CP,FUEL,SPALDG,TKE,OTHER,	CFILE155
* TITLE,PRINT,N,HF,WAR,T2,BETA,T25,FARS,EINO2C,P0,	CFILE156
* RCD,RCD2,RHC,RNOX,PT,PS,GLQC,QCO,	CFILE157
* RICH,HCHNCP,FUUL,ENTH,CONC,OTHER2	CFILE158
* ,GEX	CFILE159
504 WRITE (1)	CFILE160
* DIAJ,MJET,TIJET,VJET,PE,TE,TIE,VE,GEX,RG,PR,PRT,SC,TREF,	CFILE161
* MUREF,DIFF,NC,CNAME,ALE,SCM,CPC,NJ,NM,CT1,CT2,CT3,CT4,CT5,	CFILE162
* CT6,CT7,CT8,GJET,Y,UD,THD,TID,ED,	CFILE163
* MOLF1,MOLF2,MOLF3,MOLF4,MOLF5,MOLF6,	CFILE164
* MOLF7,MOLF8,MOLF9,MOLF10,MOLF11,MOLF12,	CFILE165
* NPIS,RAD,TS,UJ,SPV,MWT,CP,FUEL,SPALDG,TKE,OTHER,	CFILE166
* TITLE,PRINT,N,HF,WAR,T2,HFTA,T25,FARS,EINO2C,P0,	CFILE167
* RCD,RCD2,RHC,RNOX,PT,PS,HLQC,QCO,	CFILE168
* RICH,HCHNCP,FUUL,ENTH,CONC,OTHER2	CFILE169
* ,GEX	CFILE170
WRITE (6,600)	CFILE171
IF(NF,EQ,1) GO TO 1000	CFILE172
C	CFILE173
520 READ (7) KXX1,KREC,	CFILE174
* NAME,TITLE1,IDENT,ADDRES,IDENT1,TWOHT,BITS,FRK,GC,GCJ,	CFILE175
* FOOT,DIAJ,MJET,TJET,PTJET,VJET,TIJET,EJET,PE,VE,ME,TIE,TE,	CFILE176
* AXI,NJ,NM,UE,MIXPRE,XLC,FLOWJ,HERGE,NV,CON1,	CFILE177
* CT1,CT2,CT3,CT4,CT5,CT6,CT7,CT8,CT9,CTP,CTS,CTH,	CFILE178
* GAM,RG,PR,PRT,SC,TREF,MUREF,SP,SV,SLEN,DPRIN,PLOT,C6,	CFILE179
* MIX,CF,MAXIT,TDL,SUPB,	CFILE180
* X,XPRN,B,HC,TC,TIC,PTC,WJ,YJ,TTT,YSONIC,	CFILE181
* YCB,XD,RD,YR,YCD,PD,WV,MA2,VE2,TE2,NXTA,1,	CFILE182
* NC,CNAME,ALJ,ALJU,ALE,SCM,CPC,DIFF,CSC	CFILE183
WRITE (2) KXX1,KREC,	CFILE184
* NAME,TITLE1,IDENT,ADDRES,IDENT1,TWOHT,BITS,ERK,GC,GCJ,	CFILE185
* FOOT,DIAJ,MJET,TJET,PTJET,VJET,TIJET,EJET,PE,VE,ME,TIE,TE,	CFILE186
* AXI,NJ,NM,UE,MIXPRE,XLC,FLOWJ,HERGE,NV,CON1,	CFILE187
* CT1,CT2,CT3,CT4,CT5,CT6,CT7,CT8,CT9,CTP,CTS,CTH,	CFILE188
* GAM,RG,PR,PRT,SC,TREF,MUREF,SP,SV,SLEN,DPRIN,PLOT,C6,	CFILE189
* MIX,CF,MAXIT,TDL,SUPB,	CFILE190
* X,XPRN,B,HC,TC,TIC,PTC,WJ,YJ,TTT,YSONIC,	CFILE191
* YCB,XD,RD,YR,YCD,PD,WV,MA2,VE2,TE2,NXTA,1,	CFILE192
* NC,CNAME,ALJ,ALJU,ALE,SCM,CPC,DIFF,CSC	CFILE193
DO 525 IJ=1,KREC	CFILE194
READ (7) JREC,KXX,KREG,	CFILE195
* SUPD,SUPSTP,CORE,CURSTP,MER,MERSIP,NPD,	CFILE196
* PSI,Y,UD,THD,ED,TID,RMO,XLN,U,I,TOT,XMACH,	CFILE197
* PTOT,TTD,PTD,	CFILE198
* MOLF1,MOLF2,MOLF3,MOLF4,MOLF5,MOLF6,	CFILE199
* MOLF7,MOLF8,MOLF9,MOLF10,MOLF11,MOLF12,J	CFILE200

WRITE (2) JRC,KXX,KREG,	CFILF201
* SUPD,SUPSTP,CNRE,CNRSIP,MER,MERSTP,NPD,	CFILF202
* PSI,Y,UD,TND,ED,TID,RMD,XLN,J,T,TOT,XMACH,	CFILF203
* PTOT,TT,PTD,	CFILF204
* MOLF1,MOLF2,MOLF3,MOLF4,MOLF5,MOLF6,	CFILF205
* MOLF7,MOLF8,MOLF9,MOLF10,MOLF11,MOLF12,J	CFILF206
525 CONTINUE	CFILF207
WRITE (6,601)	CFILF208
C IF(NF.EQ.2) GO TO 1000	CFILF209
C	CFILF210
530 DO 535 IJ=1,KREC	CFILF211
READ (7) KXX,SY,RADS,GS,JEND	CFILF212
WRITE (3) KXX,SY,RADS,GS,JEND	CFILF213
535 CONTINUE	CFILF214
WRITE (5,602)	CFILF215
600 FORMAT(10X,22H**PREJET FILE RESTORED,10X,12HFILE CODE 1/)	CFILF216
601 FORMAT(10X,22H**JETMIX FILE RESTORED,10X,12HFILE CODE 2/)	CFILF217
602 FORMAT(10X,22H**SPALUG FILE RESTORED,10X,12HFILE CODE 3/)	CFILF218
605 FORMAT(1H1,10X,20H**FILE RESTORATION**//)	CFILF219
1000 RETURN	CFILF220
END	CFILF221
	CFILF222

COC02H	FINITE RATE CO/CO2 CHEMISTRY	COC02B01
	SUBROUTINE COC02H(P,DTIME)	COC02B02
	COMMON /PSFQ / F0A,BETS,TP,X(16),DHQDMW,TEQ,BEQ,XMWT,HNEQ	CUC02B03
	EQUIVALENCE (X1,X(1)),(X2,X(2)),(X3,X(3)),(X4,X(4)),	COC02B04
*	(X5,X(5)),(X6,X(6)),(X7,X(7)),(X8,X(8)),	CUC02B05
*	(X9,X(9)),(X10,X(10)),(X11,X(11))	COC02B06
	COMMON /GHSC / F(25),HH(25),SR(25),CPZ(25),DCPR(25)	COC02B07
	COMMON /CPRINT/ PDUM(20)	COC02B08
	COMMON /COREQN/ COREQ	COC02B09
	LOGICAL COREQ	COC02B10
	COMMON /COLIMT/ XCOLIM	COC02B11
	COMMON /PSFQX/ HQC,HUM,CD2AIR,MAIR,FS,FUELMW	COC02B12
	REAL MAIR	COC02B13
	REAL K10	COC02B14
	DATA C2/.01603286/,A10,E10/6.6E+11,1030./,TIME/0./	COC02B15
C		COC02B16
C	REACTION CO+OH = CO2+H	CUC02B17
C		CUC02B18
	TK = TP/1.8	CUC02B19
1	EQ10 = EXP(F(7)+F(5)-F(8)-F(1))	COC02B20
	IF(COREQ) GO TO 2	COC02B21
	K10 = A10*EXP(-E10/(1.98596*TK))	COC02B22
	GO TO 3	CUC02B23
2	IF(TP,LT,2250.) GO TO 21	COC02B24
	K10=1.3E+12*EXP(-5428.629857/(1.98596*TK))	COC02B25
	GO TO 3	COC02B26
21	XNUM = 1.342635076E+11*EXP(-704.02446/(1.98596*TK))	COC02B27
	XDEN = 1.-78.5243789*EXP(-13752.289/(1.98596*TK))	COC02B28
	K10 = XNUM/XDEN	COC02B29
3	XCOI = X7	COC02B30
	IF(X1,EQ.0.0,AND,X5,EQ.0.0)GO TO 20	COC02B31
	RHUM = 144.*P*C2/(1545.32*TP)	CUC02B32
	TIMEK = DTIME	COC02B33
	TIME = TIME+TIMEK	COC02B34
	TERM1 = (X1+2.*X3+X5+2.*X6)/HQC	COC02B35
	A = -RHUM*K10*(X5+X1/EQ10)	COC02B36
	B = RHUM*K10*X1/EQ10*TERM1	COC02B37
	BQA = B/A	COC02B38
10	X7 = (XCOI+BQA)*EXP(A*TIMEK)+BQA	COC02B39
	IF(X7.LE,XCOLIM) X7=XCOLIM	CUC02B40
	DXCO = XCOI-X7	COC02B41
	X8 = X8+DXCO	COC02B42
20	RETURN	COC02B43
	END	COC02B44

CCOEFF	EVALUATES COEFFICIENTS FOR -G- EQUATION	COEFF001
C		COEFF002
	SUBROUTINE COEFF(K)	COEFF003
C		COEFF004
	COMMON /CENDS/ JSTART,JEND	COEFF005
	COMMON /CPRDI/ G(50,1),ALPHA(1,1),BETA(50,1),GAMM(50,1),	COEFF006
	1 DELTA(50,1)	COEFF007
	COMMON /CJETOT/ GCJ,DTAJ,VJET,EJET,NXTA	COEFF008
	COMMON /CCONST/ CONST1,CONST2,CONST3,CONST4	COEFF009
	COMMON /CSDAT/ SY(50),RAD(50),VEL(50),TKE(50),DEN(50),TIS(50),	COEFF010
	1 FAR(50)	COEFF011
C		COEFF012
	110 DO 200 J=JSTART,JEND	COEFF013
C*	CALCULATION OF TURBULENT VISCOSITY -- $\mu_{BH}/FT-SEC$	COEFF014
	ZMUT=CONST1*DEN(J)*TIS(J)*SQRT(TKE(J))	COEFF015
	IF (J.GT.1) GO TO 120	COEFF016
C*	SYMMETRY FORCES DFQDSY TO BE ZERO ON AXIS	COEFF017
	DFQDSY=0.	COEFF018
	GO TO 130	COEFF019
	120 JP1=J+1	COEFF020
	JM1=J-1	COEFF021
C*	AT OUTER EDGE OF JET, USE ONE-SIDED DERIVATIVE	COEFF022
	IF (J.EQ.JEND) JP1=J	COEFF023
	DFQDSY=(FAR(JP1)-FAR(JM1))/(SY(JP1)-SY(JM1))	COEFF024
	130 CONTINUE	COEFF025
C*	CALCULATE COEFFICIENTS OF THE -G- EQUATION	COEFF026
	BETA(J,K)=(CONST4*ZMUT*DEN(J)*VEL(J)*RAD(J)*RAD(J)	COEFF027
	GAMM(J,K)=(CONST2*ZMUT*DEN(J)*VEL(J)*RAD(J)*RAD(J)*DFQDSY*DFQDSY	COEFF028
	IF (VEL(J).LT.(1.E-10)) GO TO 140	COEFF029
	DELTA(J,K)=-CONST3*ZMUT/(DEN(J)*VEL(J)*TIS(J)*TIS(J))	COEFF030
	GO TO 200	COEFF031
	140 DELTA(J,K)=0.	COEFF032
	200 CONTINUE	COEFF033
	RETURN	COEFF034
	END	COEFF035

*CTABPR BLOCK DATA FOR TABPRT
BLOCK DATA
COMMON /CTABPR/ IITAB
DATA IITAB/1/
END

'CTABPR' CTABPR01
CTABPR02
CTABPR03
CTABPR04
CTABPR05

CCUBSS	COEFFICIENTS FOR CUBIC SPLINE INTERPOLATION AND DIFFIN	CURSS001
	SUBROUTINE CUBS (X,Y,N)	CURSS002
C		CURSS003
	DIMENSION X(N),Y(N)	CURSS004
	DIMENSION A(40),R(40),C(40)	CURSS005
	DIMENSION S(40),T(40),U(40)	CURSS006
C		CURSS007
	COMMON/AFUN/C1,C2,C3,C4,I1PSZ,ELM,ETA,DIAM,VISC,BETA	CURSS008
C		CURSS009
	EQUIVALENC (A(1),U(1)),(R(1),T(1)),(C(1),S(1))	CURSS010
C		CURSS011
	F(X) = ((A1*X + A2)*X + A3)*X + A4	CURSS012
	DF(X) = (3.*A1*X + 2.*A2)*X + A3	CURSS013
	D2F(X) = 6.*A1*X + 2.*A2	CURSS014
C		CURSS015
	G(A) = 1./(1. - A**ETA)	CURSS016
	DG(B) = ETA/ELM*TERM** (ETA-1.)*B*B	CURSS017
	D2G(C,D,E) = C*(FTA - 1. + 2.*ETA*TERM**ETA*D)/E	CURSS018
C		CURSS019
C	FROM INPUT TABLE COMPUTE COEFFICIENTS FOR CUBIC SPLINE INTERPOLATION	CURSS020
C	AND DIFFERENTIATION	CURSS021
	C0N1 = .33333333	CURSS022
	C0N2 = .16666667	CURSS023
	DXI = X(2) - X(1)	CURSS024
	DYI = Y(2) - Y(1)	CURSS025
	DI = DYI/DXI	CURSS026
	S(1) = C0N1*DXI	CURSS027
	T(1) = C0N2*DXI	CURSS028
	U(1) = DI - (((Y(3)-Y(1))/(X(3)-X(1))) + DI)/2.	CURSS029
	NM = N-1	CURSS030
	DO 2 I=2,NM	CURSS031
	DXIM = DXI	CURSS032
	DYIM = DYI	CURSS033
	DIM = DI	CURSS034
	DXI = X(I+1) - X(I)	CURSS035
	DYI = Y(I+1) - Y(I)	CURSS036
	DI = DYI/DXI	CURSS037
	S(I) = C0N1*(DXIM + DXI)	CURSS038
	T(I) = C0N2*DXI	CURSS039
2	U(I) = DI - DIM	CURSS040
	S(N) = C0N1*DXI	CURSS041
	U(N) = (DI + (DYI+DYIM)/(DXI+DXIM))/2. - DI	CURSS042
C		CURSS043
	DO 3 I=1,NM	CURSS044
	TI = T(I)	CURSS045
	T(I) = T(I)/S(I)	CURSS046
	U(I) = U(I)/S(I)	CURSS047
	S(I+1) = S(I+1) - TI*T(I)	CURSS048
3	U(I+1) = U(I+1) - TI*U(I)	CURSS049
	U(N) = U(N)/S(N)	CURSS050

C		CUBSS051
	A(N) = U(N)	CUBSS052
	DO 4 J=1,NM	CUBSS053
	I = N-J	CUBSS054
	4 A(I) = U(I) - T(I)*A(I+1)	CUBSS055
C		CUBSS056
	DO 5 I=1,NM	CUBSS057
	DXI = X(I+1) - X(I)	CUBSS058
	DYI = Y(I+1) - Y(I)	CUBSS059
	B(I) = DYI/DXI - CON2*(A(I+1)-A(I))*DXI	CUBSS060
	5 C(I) = Y(I+1) - CON2*A(I+1)*DXI*DXI - B(I)*X(I+1)	CUBSS061
	RETURN	CUBSS062
C		CUBSS063
	ENTRY CIMP (XI,YI,DY,D2Y)	CUBSS064
	GO TO (66,10,11,11,13),ITPSZ	CUBSS065
C		CUBSS066
C	COMPUTE Y, DY/DX, D2Y/DX2 FROM CUBIC SPLINE COEFFICIENTS	CUBSS067
66	DO 6 I=1,NM	CUBSS068
	IF (X(I) .LE. XI .AND. XI .LE. X(I+1)) GO TO 7	CUBSS069
6	CONTINUE	CUBSS070
	WRITE (6,100) XI,X(I),X(N)	CUBSS071
100	FORMAT (7H0(CIMP),5X,3HXI=,F13.5,17H IS OUT OF RANGE/10X,5HX(1)=,	CUBSS072
1	F13.5,5X,5HX(N)=,F13.5)	CUBSS073
C		CUBSS074
7	DXI = X(I+1) - X(I)	CUBSS075
	A1 = (A(I+1)-A(I))/DXI/6.	CUBSS076
	AIX = A(I)*X(I+1)	CUBSS077
	AXI = A(I+1)*X(I)	CUBSS078
	A2 = (AIX - AXI)/DXI/2.	CUBSS079
	AIX = AIX*X(I+1)	CUBSS080
	AXI = AXI*X(I)	CUBSS081
	A3 = (AXI - AIX)/DXI/2. + B(I)	CUBSS082
	AIX = AIX*X(I+1)	CUBSS083
	AXI = AXI*X(I)	CUBSS084
	A4 = (AIX - AXI)/DXI/6. + C(I)	CUBSS085
C		CUBSS086
	YI = F(XI)	CUBSS087
	DY = DF(XI)	CUBSS088
	D2Y = D2F(XI)	CUBSS089
	RETURN	CUBSS090
C		CUBSS091
C	COMPUTE Y, DY/DX, D2Y/DX2 FROM INPUT POLYNOMIAL	CUBSS092
10	A1 = C1	CUBSS093
	A2 = C2	CUBSS094
	A3 = C3	CUBSS095
	A4 = C4	CUBSS096
C		CUBSS097
	YI = F(XI)	CUBSS098
	DY = DF(XI)	CUBSS099
	D2Y = D2F(XI)	CUBSS100

	RETURN	CURSS101
C		CURSS102
C	COMPUTE Y, DY/DX, D2Y/DX2 FROM INPUT SPECIAL FUNCTION	CURSS103
C	EXCEPTIONAL CASE AT X=0	CURSS104
	11 IF (XI, EQ, 0.) GO TO 12	CURSS105
	TERM = XI/FLM	CURSS106
	YI = G(TERM)	CURSS107
	DY = DG(YI)	CURSS108
	D2Y = D2G(DY, YI, XI)	CURSS109
	RETURN	CURSS110
C		CURSS111
	12 YI = 1.	CURSS112
C		CURSS113
C	FIT A CUBIC THROUGH THE POINTS (0., YI), (.05, Y2'), (.05, Y2''), AND	CURSS114
C	(.10, Y3'') IN ORDER TO FIND YI' AND YI''	CURSS115
	TERM = .05/ELM	CURSS116
	Y2 = G(TERM)	CURSS117
	Y2P = DG(Y2)	CURSS118
	Y2PP = D2G(Y2P, Y2, .05)	CURSS119
	TERM = .10/ELM	CURSS120
	Y3 = G(TERM)	CURSS121
	Y3P = DG(Y3)	CURSS122
	Y3PP = D2G(Y3P, Y3, .10)	CURSS123
C		CURSS124
	DY = (.05*(Y3PP-Y2PP)/(.10-.05)/2. - Y2PP)*.05 + Y2P	CURSS125
	D2Y = Y2PP - .05*(Y3PP-Y2PP)/(.10-.05)	CURSS126
	RETURN	CURSS127
C		CURSS128
C	V=0 CASE - ASSIGNED AREA IS NOT REQUIRED	CURSS129
	13 YI = 1.	CURSS130
	DY = 0.	CURSS131
	D2Y = 0.	CURSS132
	RETURN	CURSS133
C		CURSS134
	END	CURSS135

CDERIV	DIFF COMPATIBLE DERIVATIVE	DERIV001
	SUBROUTINE DERIV(Y,F,DFDY,J1,J2)	DERIV002
	DENOM(A,B,C)=(A-B)*(A-B)*(C-A)+(C-A)*(C-A)*(A-B)	DERIV003
	DIMENSION Y(2),F(2),DFDY(2)	DERIV004
1	DO 10 J=J1,J2	DERIV005
	DELA2=Y(J)-Y(J-1)	DERIV006
	DELB2=Y(J+1)-Y(J)	DERIV007
	DDD=1./DENOM(Y(J),Y(J-1),Y(J+1))	DERIV008
	DFDY(J)=(DELA2*DELA2*(F(J+1)-F(J))+DELB2*DELB2*(F(J)-F(J-1)))*DDD	DERIV009
10	CONTINUE	DERIV010
11	RETURN	DERIV011
	END	DERIV012

AD-A045 627

GENERAL ELECTRIC CO CINCINNATI OHIO AIRCRAFT ENGINE GROUP F/G 21/5
DEVELOPMENT OF EMISSIONS MEASUREMENT TECHNIQUES FOR AFTERBURNIN--ETC(U)
OCT 75 W C COLLEY, D R FERGUSON, M A SMITH. F33615-73-C-2047

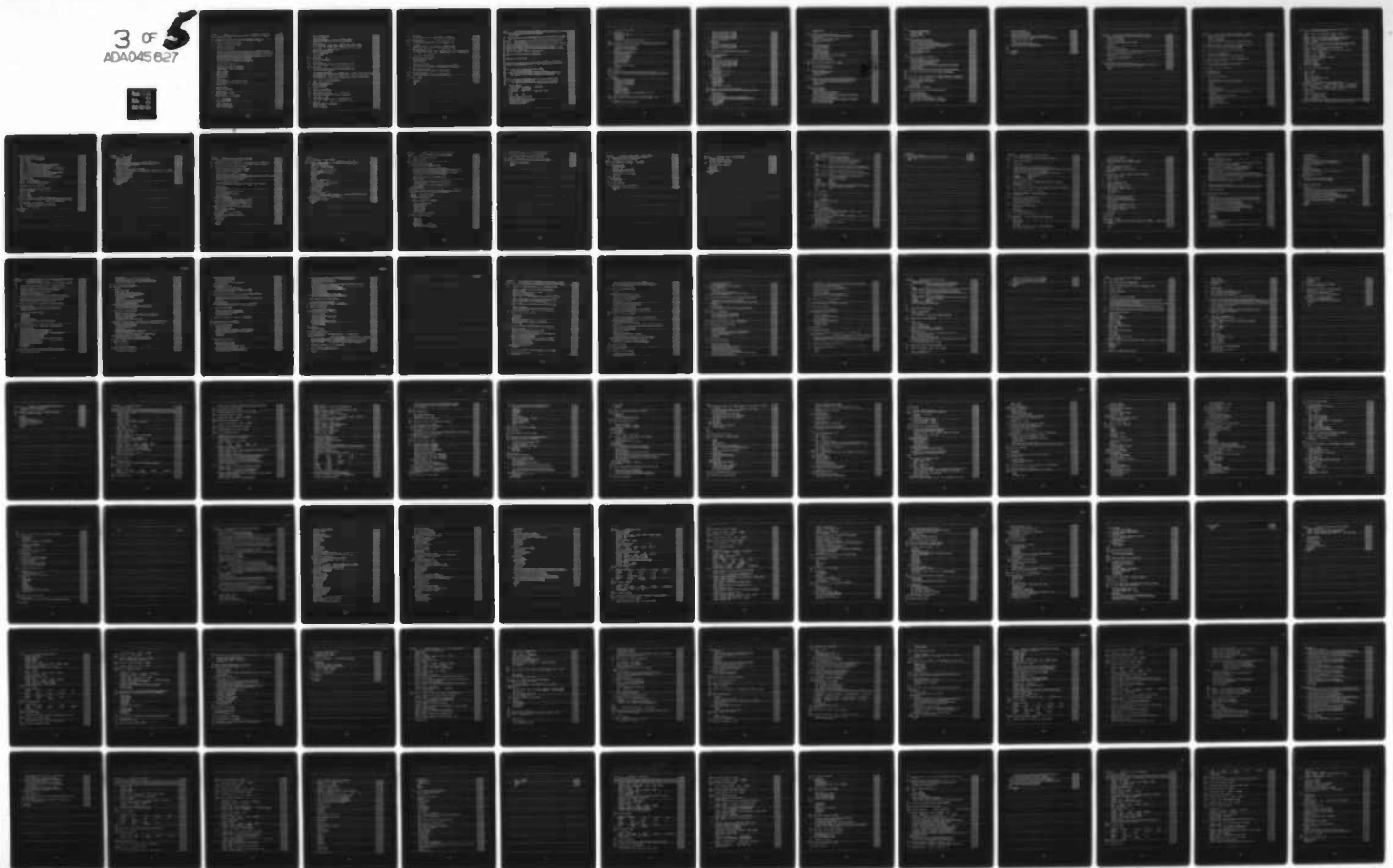
UNCLASSIFIED

R75AEG459

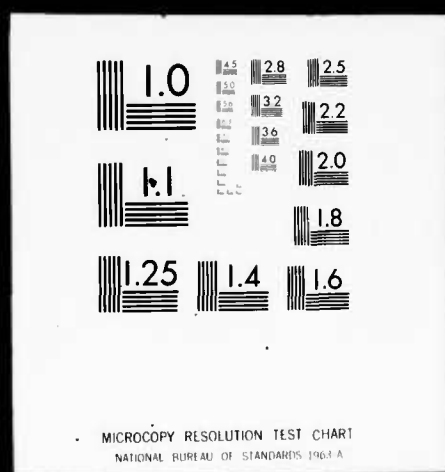
AFAPL-TR-75-52-SUPPL-2

NL

3 of 5
AD-A045 627



3 OF 5
ADA045 627



C	CDERVV	COMPUTE ALL DERIVATIVES WRT THE INDEPENDENT VARIABLE	DERVV001
C		SUBROUTINE DERV	DERVV002
C			DERVV003
C		COMPUTE ALL DERIVATIVES WRT THE INDEPENDENT VARIABLE	DERVV004
C			DERVV005
C		DOUBLE PRECISION DPS1,DPS2	DERVV006
C			DERVV007
C		LOGICAL TCON,RHOCN	DERVV008
C			DERVV009
C		REAL IVAR,MIXMW,M2	DERVV010
C			DERVV011
C		COMMON/OPTS/VERSI,TIMEV,VERSA,AREAV,FLIM,TCON,RHOCN,IPRCN	DERVV012
C		COMMON/COND/DVAR,AREA,MONT,P,IVAR,V,RHO,T,SIGMA(25),LS,LSP3,NEXT	DERVV013
C		COMMON/SPEC/SNAM(2,30),MW(25),W(25),STCIC(25,30),OMEGA(25,30)	DERVV014
C		COMMON/GHSC/GRT(25),HRT(25),SR(25),CPR(25),DCPR(25)	DERVV015
C		COMMON/DERN/F(28),ALPHA(28),BETA(28,28)	DERVV016
C		COMMON/NECC/RR,MIXMW,M2,GAMMA,TCPR,R	DERVV017
C		COMMON/SABS/S1,AA,BB,S2,DA,D2A,DTERM,IRHO	DERVV018
C			DERVV019
C		EQUIVALENC (DV,F(1))	DERVV020
C			DERVV021
C		D2ADT2(D2) = V*V*D2 + DV/V*DA	DERVV022
C		D2ADX2(D2) = (D2/V - DV*DA)/V	DERVV023
C			DERVV024
C		DPS1 = 0.00	DERVV025
C		DPS2 = 0.00	DERVV026
C		DO 1 I=1,3	DERVV027
C		F(I) = 0.	DERVV028
C		1 ALPHA(I) = 0.	DERVV029
C			DERVV030
C		DENM = RHO	DERVV031
C		IF (VERSI .NE. TIMEV) DENM = RHO*V	DERVV032
C		DO 2 I=1,LS	DERVV033
C		II = I + 3	DERVV034
C			DERVV035
C		DSIGMA/DIVAR	DERVV036
C		F(II) = W(I)/DENM	DERVV037
C			DERVV038
C		S1 FOR AA	DERVV039
C		DPS1 = DPS1 + F(II)	DERVV040
C		S2 FOR BB	DERVV041
C		2 DPS2 = DPS2 + HRT(I)*F(II)	DERVV042
C			DERVV043
C		S1 = MIXMW*DPS1	DERVV044
C		S2 = MIXMW*DPS2	DERVV045
C			DERVV046
C		GAM1 = GAMMA = 1.	DERVV047
C		BB FOR DERIVATIVES	DERVV048
C		BB = GAM1/GAMMA*S2	DERVV049
C			DERVV050

C	AA FOR DERIVATIVES	DERVV051
	AA = S1 - RB	DERVV052
	ED=GAM1*(S2-S1)	DERVV053
	IF(RHOCON.AND.V.EQ.0..AND..NOT.TCON)IRHO=2	DERVV054
C	DAVAR/DIVAR	DERVV055
C	IF (VERST .EQ. TIMEV .AND. IPRCOD .LE. 2) DA = V*DA	DERVV056
	IF (VERST .NE. TIMEV .AND. IPRCOD .GE. 3) DA = DA/V	DERVV057
C	IF (VERSA .NE. AREAV) GO TO 4	DERVV058
C	ASSIGNED AREA EQUATIONS	DERVV059
	DTERM = DA/AREA - AA	DERVV060
	T1 = 1./(M2 - 1.)	DERVV061
	T2 = M2*T1	DERVV062
C	DV/DIVAR	DERVV063
	F(1) = V*T1*DTERM	DERVV064
C	DRHO/DIVAR	DERVV065
	IF (.NOT. RHOCON) F(2) = -RHO*(T2*DTERM + AA)	DERVV066
C	DT/DIVAR	DERVV067
	IF (.NOT. TCON) F(3) = -T*(GAM1*T2*DTERM + BB)	DERVV068
	IF (IRHO.EQ.2)F(3)=-T*ED	DERVV069
C	DAREA/DIVAR WRT IVAR	DERVV070
	IF (VERST .EQ. TIMEV .AND. (IPRCOD .EQ. 1 .AND. V .NE. 0.)) D2A =	DERVV071
	* D2ADT2(D2A)	DERVV072
	IF (VERST .NE. TIMEV .AND. IPRCOD .EQ. 3) D2A = D2ADX2(D2A)	DERVV073
C	T3 = (D2A - DA*DA/AREA)/AREA	DERVV074
C	DSIGMA/DIVAR WRT IVAR	DERVV075
	DO 3 I=4,LSP3	DERVV076
	3 ALPHA(I) = 0.	DERVV077
C	DV/DIVAR WRT IVAR	DERVV078
	ALPHA(1) = V*T1*T3	DERVV079
C	DRHO/DIVAR WRT IVAR	DERVV080
	IF (.NOT. RHOCON) ALPHA(2) = -RHO*T2*T3	DERVV081
C	DT/DIVAR WRT IVAR	DERVV082
	IF (.NOT. TCON) ALPHA(3) = -T*GAM1*T2*T3	DERVV083
C	GO TO 6	DERVV084
C	ASSIGNED PRESSURE EQUATIONS	DERVV085
	4 DTERM = DA/P	DERVV086
	T2 = -1./GAMMA	DERVV087
C		DERVV088
		DERVV089
		DERVV090
		DERVV091
		DERVV092
		DERVV093
		DERVV094
		DERVV095
		DERVV096
		DERVV097
		DERVV098
		DERVV099
		DERVV100

C	DV/DIVAR	DERVV101
	IF (V.NE. 0.) F(1) = -DA/(RHO*V)*1.01325E+06	DERVV102
C		DERVV103
C	DRHO/DIVAR	DERVV104
	IF (.NOT. RHOCON) F(2) = -RHO*(T2*DTerm + AA)	DERVV105
C		DERVV106
C	DT/DIVAR	DERVV107
	IF (.NOT. TCON) F(3) = -T*(GAM1*T2*DTerm + BB)	DERVV108
C		DERVV109
C	DP/DIVAR WRT IVAR	DERVV110
	IF (VERSI.EQ. TIMEV .AND. IPRCOD.EQ. 2) D2A = D2ADT2(D2A)	DERVV111
	IF (VERSI.NE. TIMEV .AND. IPRCOD.EQ. 4) D2A = D2ADX2(D2A)	DERVV112
C		DERVV113
	T3 = (D2A - DA*DA/P)/P	DERVV114
C		DERVV115
C	DSIGMA/DIVAR WRT IVAR	DERVV116
	DO 5 I=4,LSP3	DERVV117
	5 ALPHA(I) = 0.	DERVV118
C		DERVV119
C	DV/DIVAR WRT IVAR	DERVV120
	IF (V.NE. 0.) ALPHA(1) = -D2A/(RHO*V)*1.01325E+06	DERVV121
C		DERVV122
C	DRHO/DIVAR WRT IVAR	DERVV123
	IF (.NOT. RHOCON) ALPHA(2) = -RHO*T2*T3	DERVV124
C		DERVV125
C	DT/DIVAR WRT IVAR	DERVV126
	IF (.NOT. TCON) ALPHA(3) = -T*GAM1*T2*T3	DERVV127
C		DERVV128
	6 CALL PARD	DERVV129
C		DERVV130
	RETURN	DERVV131
	END	DERVV132

COFEQ	GEN. ROUTINE-DIFFUSION EQUATION EON==CENTERED/NNN CENTERED	DFEQ0001
	SUBROUTINE DFEG(NDIFF,NBCL,NBCU,IERIND)	DFEQ0002
	LOGICAL FIRST	DFEQ0003
C*		DFEQ0004
	DENOM(AA,BB,CC)=(AA-BB)*(AA-BB)*(CC-AA)*(CC-AA)*(CC-AA)*(AA-BB)	DFEQ0005
C*		DFEQ0006
C*	PDEQ== EPSLN*(DF/DX)=ALPHA*D(BETA*(DF/DPHI))/DPHI+GAMMA+DLTA*F	DFEQ0007
C*	+ETA*(DF/DPHI)	DFEQ0008
C*	FMI=3OLUTION VECTOR==PREVIOUS STATION DESTROYED BY CURRENT STATION	DFEQ0009
C*	DFMI= DERIVATIVE VECTOR	DFEQ0010
C*	ALPHA,BETA,GAMMA,DLTA,EPSLN,ETA==COEFFICIENTS IN PDEQ	DFEQ0011
C*	DELM1,DELM2= NORMAL STEP SIZES AT PREVIOUS AND CURRENT STATIONS	DFEQ0012
C*	DELX= AXIAL STEP SIZE	DFEQ0013
C*	B1,C1,D1,AN,BN,DN= BOUNDARY COEFFICIENTS	DFEQ0014
C		DFEQ0015
C*	USES NON CENTERED 3-POINT DIFFERENCES	DFEQ0016
C*		DFEQ0017
C*		DFEQ0018
C*	STANDARD IMPLICIT FORM	DFEQ0019
C*		DFEQ0020
C*	BRANCH FOR DISCONTINUOUS FUNCTION FORM REQUIRES INTERPOLATION	DFEQ0021
C*	AT EACH MESH POINT....	DFEQ0022
C*		DFEQ0023
C*		DFEQ0024
C*		DFEQ0025
C*	NBC==NORMAL BC INDICATOR==L=LOWER,U=UPPER	DFEQ0026
C*	=0 BACKWARD DIFFERENCE	DFEQ0027
C*	=1 USE TWO POINTS ABOVE OR BELOW BOUNDARY	DFEQ0028
C*	EXTRA COEFFICIENTS STORED IN CARRY(1,1) AND/OR CARRY(3,NM)	DFEQ0029
C*	=2 USE R.C. WHICH SATISFY DIFFERENTIAL EQUATION	DFEQ0030
C*		DFEQ0031
C*		DFEQ0032
C*	NDIFF==INDICATES DIFFERENCE FORM FOR D(BETA*DF/DPHI)/DPHI	DFEQ0033
C*	=0 EXPAND 2-ND DERIVATIVE	DFEQ0034
C*	=1 USE FORM APPLICABLE FOR DISCONTINUOUS FUNCTIONS	DFEQ0035
C*		DFEQ0036
	COMMON /PARAM/	DFEQ0037
	* ALPHA(200) , BETA(200) , GAMMA(200) ,	DFEQ0038
	* EPSLN(200) , DLTA(200) ,	DFEQ0039
	* FMI(200) , DFMI(200) ,	DFEQ0040
	* PHI1(200) , NM1 , PHI2(200) , NM ,	DFEQ0041
	* DX ,	DFEQ0042
	* B1 , C1 , D1 ,	DFEQ0043
	* AN , BN , DN	DFEQ0044
	COMMON /PARAM1/ ETA(200)	DFEQ0045
	COMMON /YOFXI/ NLC	DFEQ0046
	EQUIVALENCE (PHI(1),PHI2(1))	DFEQ0047
	DIMENSION CARRY(200,4)	DFEQ0048
	DIMENSION PHI(200)	DFEQ0049
C*		DFEQ0050

	DATA FIRST/T/,NUM/200/	DFEQ0051
C*		DFEQ0052
1	IERIND=0	DFEQ0053
	IF(DX.EQ.0.) DX=1.	DFEQ0054
	NGOL=NBCL+1	DFEQ0055
	NGOU=NRCU+1	DFEQ0056
	NDF=NDIFF+1	DFEQ0057
	IF(PHI2(NM).LE.PHI1(NM)) GO TO 10	DFEQ0058
C*		DFEQ0059
C*	ADD POINTS TO PREVIOUS STATION TO ACCOUNT FOR MESH SPREADING	DFEQ0060
C*		DFEQ0061
	DELM1=PHI1(NM1)-PHI1(NM1-1)	DFEQ0062
	DELTA=PHI2(NM)-PHI1(NM1)	DFEQ0063
	NEND=2+FIX(DELTA/DELM1)+NM1	DFEQ0064
	J=NM1+1	DFEQ0065
5	PHI1(J)=PHI1(J-1)+DELM1	DFEQ0066
	ALPHA(J)=ALPHA(NM1)	DFEQ0067
	BETA(J)=BETA(NM1)	DFEQ0068
	GAMMA(J)=GAMMA(NM1)	DFEQ0069
	EPSLN(J)=EPSLN(NM1)	DFEQ0070
	DLTA(J)=DLTA(NM1)	DFEQ0071
	ETA(J)=ETA(NM1)	DFEQ0072
6	FM1(J)=FM1(NM1)	DFEQ0073
	J=J+1	DFEQ0074
	IF(J.LE.NEND) GO TO 5	DFEQ0075
	NM1=NEND	DFEQ0076
C*		DFEQ0077
10	NMM1=NM-1	DFEQ0078
C*		DFEQ0079
C*	CONSTRUCT COEFFICIENT ARRAY FOR NODES 2 TO (NMM1)	DFEQ0080
C*	DELTA=S AT EACH STATION ARE SAME, BYPASS INTERPOLATION SECTION	DFEQ0081
C*		DFEQ0082
12	J=1	DFEQ0083
13	J=J+1	DFEQ0084
	DEL1=PHI1(J)-PHI1(J-1)	DFEQ0085
	DEL2=PHI1(J+1)-PHI1(J)	DFEQ0086
	DEL3=PHI2(J)-PHI2(J-1)	DFEQ0087
	DEL4=PHI2(J+1)-PHI2(J)	DFEQ0088
	Y=PHI2(J)	DFEQ0089
	DEL34=DEL3+DEL4	DFEQ0090
	DEL35Q=DEL3+DEL3	DFEQ0091
	DEL45Q=DEL4+DEL4	DFEQ0092
	ASSIGN 21 TO KGO	DFEQ0093
	IF(DEL1.EQ.DEL3 .AND. DEL2.EQ.DEL4) GO TO 19	DFEQ0094
	ASSIGN 37 TO KGO	DFEQ0095
C*		DFEQ0096
C*	LINEAR INTERPOLATION FOR ALP,GAM,FM2,EPS,BETA ETC	DFEQ0097
C*		DFEQ0098
	NLC=1	DFEQ0099
14	ALP=YOF(Y,PHI1,ALPHA,1,NM1)	DFEQ0100

GAH=YOF(Y,PHI1,GAMMA,1,NM1)	DFEQ0101
FM2=YOF(Y,PHI1,FM1,1,NM1)	DFEQ0102
EPS=YOF(Y,PHI1,EPSLN,1,NM1)	DFEQ0103
DLTA1=YOF(Y,PHI1,DLTA,1,NM1)	DFEQ0104
ETA1=YOF(Y,PHI1,ETA,1,NM1)	DFEQ0105
GO TO (15,16),NDF	DFEQ0106
15 W=Y+DEL4	DFEQ0107
Z=Y+DEL3	DFEQ0108
BETL=YOF(Z,PHI1,BETA,1,NM1)	DFEQ0109
BET=YOF(Y,PHI1,BETA,1,NM1)	DFEQ0110
BETU=YOF(W,PHI1,BETA,1,NM1)	DFEQ0111
GO TO 30	DFEQ0112
C* DISCONTINUOUS FUNCTION(BETA*DF/DPHI) BRANCH	DFEQ0113
C*	DFEQ0114
C*	DFEQ0115
19 GO TO (20,16),NDF	DFEQ0116
16 W=Y+.5*DEL4	DFEQ0117
Z=Y+.5*DEL3	DFEQ0118
NLC=1	DFEQ0119
BMHALF=YOF(Z,PHI1,BETA,1,NM1)	DFEQ0120
FMHALF=YOF(Z,PHI1,ETA,1,NM1)	DFEQ0121
BPHALF=YOF(W,PHI1,BETA,1,NM1)	DFEQ0122
EPHALF=YOF(W,PHI1,ETA,1,NM1)	DFEQ0123
IF(J.NE.2) GO TO 161	DFEQ0124
B3V=BMHALF	DFEQ0125
E3V=FMHALF	DFEQ0126
161 IF(J.NE. NM1) GO TO 20	DFEQ0127
B3VU=BPHALF	DFEQ0128
E3VU=EPHALF	DFEQ0129
20 GO TO KGT, (21,37)	DFEQ0130
21 ALP=ALPHA(J)	DFEQ0131
GAM=GAMMA(J)	DFEQ0132
EPS=EPSLN(J)	DFEQ0133
DLTA1=DLTA(J)	DFEQ0134
ETA1=ETA(J)	DFEQ0135
FM2=FM1(J)	DFEQ0136
IF(NDF.EQ.2) GO TO 37	DFEQ0137
C*	DFEQ0138
C* EXPANDED 2-ND DERIVATIVE BRANCH----	DFEQ0139
C* D(BETA*DF/DPHI)/DPHI=DBETA/DPHI*DF/DPHI+BETA*D2P/DPHI2	DFEQ0140
C*	DFEQ0141
BET=BETA(J)	DFEQ0142
BETU=BETA(J+1)	DFEQ0143
BETL=BETA(J-1)	DFEQ0144
30 DD=1./DENOM(PHI2(J),PHI2(J-1),PHI2(J+1))	DFEQ0145
DBETA=(DEL390*(BETU-BET)+DEL490*(BET-BETL))*DD	DFEQ0146
32 TSV=ALP*DX*DD	DFEQ0147
TL=2.*TSV*BET	DFEQ0148
TM=TSV*DBETA	DFEQ0149
TN=GAM*DX	DFEQ0150

TK=ETA1*DX*DD	DFEQ0151
TMK=TK+TM	DFEQ0152
C* 35 CARRY(J,1)=TMK*DEL4SQ-TL*DEL4	DFEQ0153
CARRY(J,2)=EPS+TL*DEL34+TMK*(DEL3SQ-DEL4SQ)	DFEQ0154
*=DLTA1*DX	DFEQ0155
CARRY(J,3)=TMK*DEL3SQ-TL*DEL3	DFEQ0156
CARRY(J,4)=EPS*FM2+TN	DFEQ0157
GO TO 39	DFEQ0158
C* 37 TSV=2.*ALP*DX/DEL34	DFEQ0159
TL=TSV*BMHALF/DEL4	DFEQ0160
TM=TSV*BMHALF/DEL3	DFEQ0161
TN=GAM*DX	DFEQ0162
TKM=.5*EMHALF/DEL3	DFEQ0163
TKP=.5*EPHALF/DEL4	DFEQ0164
CARRY(J,1)=TM+TKM	DFEQ0165
CARRY(J,2)=EPS+TM+TL*DLTA1*DX+TKP*TKM	DFEQ0166
CARRY(J,3)=TL+TKP	DFEQ0167
CARRY(J,4)=EPS*FM2+TN	DFEQ0168
39 IF(J.LT.NMM1) GO TO 13	DFEQ0169
C* 39 STORE UPPER AND LOWER B.C. IN CARRY	DFEQ0170
C* DELT1=PHI(2)-PHI(1)	DFEQ0171
GO TO (40,46,400),NGOL	DFEQ0172
C* 40 CARRY(1,1)=0.	DFEQ0173
CARRY(1,2)=B1=C1/DELT1	DFEQ0174
CARRY(1,3)=C1/DELT1	DFEQ0175
CARRY(1,4)=D1	DFEQ0176
GO TO 47	DFEQ0177
C* 46 CARRY(1,1)=C1*PHI(2)/(PHI(3)*(PHI(3)-PHI(2)))	DFEQ0178
CARRY(1,2)=B1=(PHI(2)+PHI(3))/(PHI(2)*PHI(3))*C1	DFEQ0179
CARRY(1,3)=CARRY(1,1)*PHI(3)**2/PHI(2)**2	DFEQ0180
CARRY(1,4)=D1	DFEQ0181
GO TO 47	DFEQ0182
C* 400 DELT2=1./DELT1	DFEQ0183
DELT13=DFLT2**2	DFEQ0184
CARRY(1,1)=0.	DFEQ0185
TERM=2.*C1*ALPHA(1)*BSV*DELT13	DFEQ0186
TERM2=2.*D1*ALPHA(1)*DELT2	DFEQ0187
CARRY(1,2)=2.*B1*ALPHA(1)*DELT2*TERM-C1*EPSLN(1)/DX	DFEQ0188
*+C1*DLTA(1)-C1*FSV*DELT1	DFEQ0189
CARRY(1,3)=TERM+C1*FSV*DELT1	DFEQ0190
CARRY(1,4)=C1*(GAMMA(1)+FM1(1)*EPSLN(1)/DX)+TERM2	DFEQ0191
47 DELTN=PHI(NM1)-PHI(NMM1)	DFEQ0192
GO TO (48,49,410),NGOU	DFEQ0193
	DFEQ0194
	DFEQ0195
	DFEQ0196
	DFEQ0197
	DFEQ0198
	DFEQ0199
	DFEQ0200

C*		DFEQ0201
48	CARRY(NM,1)=BN/DELTA	DFEQ0202
	CARRY(NM,2)=AN+BN/DELTA	DFEQ0203
	CARRY(NM,3)=0.	DFEQ0204
	CARRY(NM,4)=DN	DFEQ0205
	GO TO 50	DFEQ0206
C*		DFEQ0207
49	PHI1=PHI(NMM1)*PHI(NM1)	DFEQ0208
	PHI2=PHI(NM2)*PHI(NM2)	DFEQ0209
	ZTA=(2.*PHI(NM)*PHI(NMM1)-PHI1)/	DFEQ0210
	*(PHI2+2.*PHI(NM)*(PHI(NMM1)-PHI(NM2))-PHI1)	DFEQ0211
	DEN=1./((1.-ZTA)*PHI(NMM1)+ZTA*PHI(NM2))	DFEQ0212
	CARRY(NM,1)=(1.-ZTA)*BN+DEN	DFEQ0213
	CARRY(NM,2)=AN+BN+DEN	DFEQ0214
	*BN*DLTA(NM)	DFEQ0215
	CARRY(NM,3)=ZTA*BN+DEN	DFEQ0216
	CARRY(NM,4)=DN	DFEQ0217
	GO TO 50	DFEQ0218
C*		DFEQ0219
410	DELTA=1./((PHI(NM)-PHI(NMM1)))	DFEQ0220
	DELTA=DELTA**2	DFEQ0221
	TERM=2.*BN*ALPHA(NM)+BSVU*DELTA	DFEQ0222
	TERM2=2.*DN*ALPHA(NM)*DELTA	DFEQ0223
	CARRY(NM,1)=TERM+BN+ESVU*DELTA	DFEQ0224
	CARRY(NM,2)=2.*AN*ALPHA(NM)*DELTA+BN+EPSLN(NM)/DX+TERM	DFEQ0225
	*BN+ESVU*DELTA	DFEQ0226
	CARRY(NM,3)=0.	DFEQ0227
	CARRY(NM,4)=BN*(GAMMA(NM)+EPSLN(NM)+FMI(NM)/DX)+TERM2	DFEQ0228
C*		DFEQ0229
C*	CALL TDSQ FOR SOLUTION TO SIMULTANEOUS EQUATIONS.	DFEQ0230
C*		DFEQ0231
C*		DFEQ0232
C*	FIRST CALL SETS UP ADDRESSES FOR TDSQ...BYPASS ON SUBSEQUENT	DFEQ0233
C*		DFEQ0234
	50 IF(.NOT. FIRST) GO TO 51	DFEQ0235
	CALL TDSQ(CARRY,NM,NUM,ERR)	DFEQ0236
	FIRST=.FALSE.	DFEQ0237
	GO TO 52	DFEQ0238
	51 CALL TDSQ1	DFEQ0239
	52 IF(ERR.GT. 0.) IERIND=1	DFEQ0240
C*		DFEQ0241
C*	STORE SOLUTION IN FMI AND COMPUTE DF/DPHI	DFEQ0242
C*		DFEQ0243
	FMI(1)=CARRY(1,1)	DFEQ0244
	FMI(NM)=CARRY(NM,1)	DFEQ0245
	DELTA=PHI(2)-PHI(1)	DFEQ0246
	DELTA=PHI(NM)-PHI(NMM1)	DFEQ0247
	IF(B1.EQ.0..AND.D1.EQ.0.) DFMI(1)=0.	DFEQ0248
	IF(AN.EQ.0..AND.DN.EQ.0.) DFMI(NM)=0.	DFEQ0249
	J=2	DFEQ0250

61	FM1(J)=CARRY(J,1)	DFEQ0251
	FM1(J+1)=CARRY(J+1,1)	DFEQ0252
	DELA2=PHI(J)-PHI(J-1)	DFEQ0253
	DELB2=PHI(J+1)-PHI(J)	DFEQ0254
	DDD=1./DENOM(PHI2(J),PHI2(J-1),PHI2(J+1))	DFEQ0255
	DFM1(J)=(DELA2*DELA2*(FM1(J+1)-FM1(J))+DELB2*DELB2	DFEQ0256
	* *(FM1(J)-FM1(J-1)))*DDD	DFEQ0257
	J=J+1	DFEQ0258
	IF(J.LE.NMM1) GO TO 61	DFEQ0259
	DFM1(1)=(FM1(2)-FM1(1))/DEL T1	DFEQ0260
	DFM1(NM)=(FM1(NM)-FM1(NMM1))/DEL TN	DFEQ0261
C*		DFEQ0262
C*		DFEQ0263
C*		DFEQ0264
	NM1=NM	DFEQ0265
100	RETURN	DFEQ0266
	END	DFEQ0267

CEQGASH	THEIRMO PRPTYS OF EQLHRM COMBN GAS - H INDEP.	EQGASH01
	SUBROUTINE EQGASH(FAR,WAR,HC,I,P,FXCD,FXND,Z,	EQGASH02
	H,MWT,S,SPV,A,CP)	EQGASH03
	REAL MWT,Z(11),QV(8)	EQGASH04
	LOGICAL FXCD,FXND	EQGASH05
	COMMON/COIREM/HTOL,H0,DHDT,CTRMAX	EQGASH06
	HTOL=0.01	EQGASH07
	H0=H	EQGASH08
	QV(1)=0.0	EQGASH09
10	IF(1,LT,189.0)T=189.0	EQGASH10
	IF(1,GT,9000.0)T=9000.0	EQGASH11
	CALL EQGAST(FAR,WAR,HC,I,P,FXCD,FXND,Z,H1,MWT,S,SPV,A,CP)	EQGASH12
	IF((T,EQ,189.0).AND.(H1,GT,H0))GO TO 20	EQGASH13
	IF(QV(1),EQ,25.0)GO TO 20	EQGASH14
	DHDT=CP	EQGASH15
	CALL QIREM(T,H1,-5000.0,QV)	EQGASH16
	IF(QV(1),NE,0.0)GO TO 10	EQGASH17
	RETURN	EQGASH18
C		EQGASH19
20	WRITE(6,30)T,H1,H0,CP,FAR	EQGASH20
30	FORMAT(27HCEQGASH - NOT CONVERGED. T=,F7.2,4H H=,F8.2,	EQGASH21
	15H H0=,F8.2,5H CP=,F8.5,6H FAR=,F9.6)	EQGASH22
	RETURN	EQGASH23
	END	EQGASH24

CEQGAST	THERMO PRPTYS OF EQLDRM COMBN GAS - T INDEP.	EQGAST01
	SUBROUTINE EQGAST(FAR,WAR,HC,T,P,FXCO,FXND,Z,	EQGAST02
	H,MWT,S,SPV,A,CP)	EQGAST03
	REAL MWT,7(11),JJ	EQGAST04
	LOGICAL FIXCO,FXND	EQGAST05
	COMMON/GHSC/FF(25),HH(25),SS(25),CCP(25),DCPDY(25)	EQGAST06
C		EQGAST07
C	ORDER OF SPECIES = H ₂ O,H ₂ ,O ₂ ,NH ₃ ,H ₂ O,CO,CO ₂ ,N ₂ ,A,N ₂	EQGAST08
C	IF FIXCO, 7(CO) AND Z(CO ₂) MUST BE PRESET	EQGAST09
C	IF FXND, Z(N ₂) AND Z(N ₂) MUST BE PRESET	EQGAST10
C		EQGAST11
C	GET EQUILIBRIUM COMPOSITION (MOLES/LB MIX)	EQGAST12
C		EQGAST13
	IF (FAR.LT.0.0) FAR=0.0	EQGAST14
	IF (WAR.LE.0.0) WAR=1E-8	EQGAST15
	CALL MCEQ2(FAR,WAR,HC,T,P,FXCO,FXND,Z)	EQGAST16
C		EQGAST17
C	ENTRY FOR DETERMINATION OF THERMO PROPERTIES OF	EQGAST18
C	PREDETERMINED MIXTURES.	EQGAST19
C		EQGAST20
	ENTRY FRGAST(T,P,7,H,MWT,S,SPV,A,CP)	EQGAST21
	CALL THRM(T/1.8,1.0)	EQGAST22
C		EQGAST23
C	SPECIFIC MOLARITY (1/MWT)	EQGAST24
C		EQGAST25
	10 SPM=0.0	EQGAST26
	DO 20 I=1,11	EQGAST27
	20 SPM=SPM+7(I)	EQGAST28
C		EQGAST29
C	THERMOSTATIC PROPERTIES	EQGAST30
C		EQGAST31
	MWT=1.0/SPM	EQGAST32
	DATA R0,G0,JJ/1.98596,32.1740,778.2/	EQGAST33
	CP=0.0	EQGAST34
	H=0.0	EQGAST35
	S=0.0	EQGAST36
	DO 30 I=1,11	EQGAST37
	CP=CP+7(I)*CCP(I)	EQGAST38
	H=H+7(I)*HH(I)	EQGAST39
30	S=S+7(I)*(HH(I)-FF(I)-ALOG(CAMAX1(Z(I),1E-38)))	EQGAST40
	CP=CP+R0	EQGAST41
	H=H+R0*T	EQGAST42
	S=(S-SPM*ALOG(MWT*P/14.696))*R0	EQGAST43
	PSF=144.0*P	EQGAST44
	SPV=R0*J.IAT*SPM/PSF	EQGAST45
	GAM=CP*MWT/R0	EQGAST46
	GAM=GAM/(GAM-1.0)	EQGAST47
	A=SQRT(PSF*SPV*GAM*G0)	EQGAST48
	RETURN	EQGAST49
	END	EQGAST50

CEQKIN	EQUILIBRIUM TERMS FOR KINETICS CALCULATION	EQKIN001
	SUBROUTINE EQKIN(P,H,FXCO)	EQKIN002
C		EQKIN003
	COMMON /PSEQX / HOC,HUM,CO2AIR,MAIR,FS,FUELMW	EQKIN004
	REAL MAIR	EQKIN005
	COMMON /GHSC / FF(25),HZ(25),SR(25),CPZ(25),DCPR(25)	EQKIN006
	COMMON /CEQKIN/ BK,ALK,DELHNO,DEN,XMWEQ,XNO,XCO	EQKIN007
	COMMON /KININS/ XMUN,XMWC,TCONST,CONER	EQKIN008
	LOGICAL TCONST	EQKIN009
	COMMON /PSEQ / FOA,BETS,TP,X(16),DHQDMW,TEQ,BEQ,XMWT,MNEQ	EQKIN010
	EQUIVALENCE (X1,X(1)),(X2,X(2)),(X3,X(3)),(X4,X(4)),	EQKIN011
	*(X5,X(5))	EQKIN012
	COMMON /PRQPR / PR,HHR,TR,FCR,RHOR,RR,WMWR	EQKIN013
	COMMON /CBITS / BITS,BLANK	EQKIN014
	COMMON /CKINET/ DUMK(89),RK1,RK2,RK3,RK4,RK5,TK,ENTRY1	EQKIN015
	LOGICAL ENTRY1	EQKIN016
	COMMON /CCNCO2/ CORATE,XCOI	EQKIN017
	LOGICAL CORATE	EQKIN018
	COMMON /CPRINT/ PDUM(20)	EQKIN019
	COMMON /SNMW / ALSP(150), WMT(75)	EQKIN020
	COMMON/DJIREM/KINFIL	EQKIN021
	LOGICAL FIXCO	EQKIN022
	DIMENSION QV(8)	EQKIN023
	DATA R0/1.98596/	EQKIN024
	DATA C2/.01603286/	EQKIN025
C		EQKIN026
C	GET EQUILIBRIUM BETA,TEMP,ETC.	EQKIN027
C		EQKIN028
	X(7) = XCO	EQKIN029
	X(11) = XNO	EQKIN030
	HH = H	EQKIN031
	TEQ1 = TP	EQKIN032
	XJP = 0.1*TEQ1	EQKIN033
	IF(PDUM(20),LE,0.0)GO TO 10	EQKIN034
	KINFIL=PDUM(20)+0.1	EQKIN035
	REWIND KINFIL	EQKIN036
	WRITE(KINFIL,1001)P,HH,FOA,HUM,HOC,XCO,XNO	EQKIN037
1001	FORMAT(45H **EQKIN** TEMPERATURE ITERATION HISTORY,87X/	EQKIN038
	'3H P=,F8.4,6H H=,F10.4,8H FOA=,F9.6,8H HUM=,F9.6,	EQKIN039
	'8H H/C=,F6.3,8H XCO=,E13.6,8H XNO=,E13.6,15X/132X)	EQKIN040
10	CALL HCEQ2(FOA,HUM,HOC,TEQ1,P,FXCO,.TRUE., X)	EQKIN041
	XMWEQ = 0.	EQKIN042
	HH1 = 0.	EQKIN043
	DO 12 K=1,11	EQKIN044
	XMWEQ = XMWEQ+X(K)*WMT(K)	EQKIN045
12	HH1 = HH1+X(K)*HZ(K)	EQKIN046
	HH1 = R0*TEQ1*HH1/XMWEQ	EQKIN047
	ERR = HH-HH1	EQKIN048
	IF(PDUM(20),GT,0.0)WRITE(KINFIL,1002)TEQ1,XMWEQ,HH1,(X(I),I=1,12)	EQKIN049
1002	FORMAT(3H T=,F10.4,5H MW=,F10.6,5H H1=,F10.4,27X,2HX=,	EQKIN050

14E15.8/12X,8E15.8)	EQKIN051
CALL QIREM(TEQ1,ERR,XJP,QV)	EQKIN052
IF(QV(1),NE,0.) GO TO 10	EQKIN053
KINFIL=40	EQKIN054
TEQ = TEQ1	EQKIN055
GAR = 1./FOA+HUM	EQKIN056
SUMH = (FOA*HQC/FUELMW+HUM*2./18.016)/GAR	EQKIN057
SUMC = (FOA/FUELMW+CO2AIR/MAIR)/GAR	EQKIN058
SUMD = (HUM/18.016+2.*(0.209495+CO2AIR)/MAIR)/GAR	EQKIN059
SUMN = 2.*.780881/(MAIR*GAR)	EQKIN060
SUMA = (.009624-CO2AIR)/(MAIR*GAR)	EQKIN061
ZZX = .5*(CO2AIR+MAIR/18.016+HUM)*SUMN/.780881	EQKIN062
SPMU = .5*(SUMD+SUMC+SUMH+SUMN+ZZX)+SUMA	EQKIN063
SPMC = .5*(.5*SUMH+SUMN+SUMD)+SUMA	EQKIN064
IF(FOA,GE,FS) SPMC=.5*(SUMH+SUMN)+SUMC+SUMA	EQKIN065
XMWUN = 1./SPMU	EQKIN066
XMWC = 1./SPMC	EQKIN067
DEN = 1./((SPMC-SPMU)	EQKIN068
BEO = (1./XMWEO-1./XMWUN)*DEN	EQKIN069
C	EQKIN070
C CALCULATE TERMS FOR KINETICS	EQKIN071
C	EQKIN072
RHOS = 144.*XMWEO*P/(1545.32*TEQ)	EQKIN073
RHOCGS= C2*RHOS	EQKIN074
TK = TEQ/1.8	EQKIN075
TEMP = TP	EQKIN076
TP = TEQ	EQKIN077
CALL RATCON(P)	EQKIN078
60 TP = TEMP	EQKIN079
RATE = -RHOCGS**2/XMWEO**3*(RK1*X1*X5+RK2*X1**2+RK3*X1*X2+	EQKIN080
* RK4*X2**2+RK5*X1*X4)	EQKIN081
BK = 0.	EQKIN082
IF(BEO,GE,1.) GO TO 100	EQKIN083
B = RATE/((1./XMWEO-1./XMWC)**2*DEN)	EQKIN084
BK = B*(1.-BEO)**2	EQKIN085
C	EQKIN086
100 RETURN	EQKIN087
END	EQKIN088

CERROR1	QIREM QUITs	ERROR101
	SUBROUTINE ERROR1	ERROR102
	LOGICAL NODATA	ERROR103
	DIMENSION A(22)	ERROR104
	COMMON/CQIREM/YTOL,YO,DYDX,TUMANY/DOIREM/NFILE	ERROR105
	CALL FLGEOF(NFILE,NODATA)	ERROR106
	END FILE NFILE	ERROR107
	REWIND NFILE	ERROR108
	WRITE(6,100)YTOL,YO,DYDX,TUMANY	ERROR109
100	FORMAT(6H0YTOL=,E16.8,5H YO=,E16.8,7H DYDX=,F16.8,9H TUMANY=,	ERROR110
	1E16.8)	ERROR111
	DO 10 LINES=1,3000	ERROR112
	READ(NFILE,110)A	ERROR113
	IF(NODATA)GO TO 20	ERROR114
10	WRITE(6,110)A	ERROR115
110	FORMAT(22A6)	ERROR116
20	CALL ERROR	ERROR117
	RETURN	ERROR118
	END	ERROR119

C	ERRORC	RELATIVE ERROR IN INTEGRATION STEP	ERRORC01
		FUNCTION ERRORC(Y,RK,E,JC,H)	ERRORC02
C			ERRORC03
C		COMPUTE THE RELATIVE ERROR IN AN INTEGRATION STEP OF SIZE H	ERRORC04
C			ERRORC05
C		DETERMINE THE CONTROLLING VARIABLE	ERRORC06
C			ERRORC07
		DOUBLE PRECISION DNP1,DN,DNM1,C,ABSY	ERRORC08
C			ERRORC09
C		DIMENSION Y(28),RK(28),E(28)	ERRORC10
C			ERRORC11
		COMMON/CONN/DUM1(33),LS,LSP3,NEXT	ERRORC12
		COMMON/STNT/HMTN,HNM1,HN,HNP1,HMAX,NH,AVH,EMAX,ERRN,JC,VC,KOUNT,ERRP	ERRORC13
		COMMON/PQRF/PK(28),QK(28),DUM2(56)	ERRORC14
		COMMON/SKIP/NEGL(25),I1,I2,I3	ERRORC15
C			ERRORC16
		C = (H + HN)/(HN + HNM1)	ERRORC17
C		FAC1 = ABS(H*H/(2.*H + HN)*(2.*H + HN)/(H + HN + HNM1))	ERRORC18
			ERRORC19
		ERRORC = -1.	ERRORC20
		DO 4 I=1,LSP3	ERRORC21
		E(I) = 0.	ERRORC22
		IF (ABS(RK(I)) .EQ. 0.) GO TO 4	ERRORC23
		ABSY = ABS(Y(I))	ERRORC24
		IF (ABSY .EQ. 0.) GO TO 4	ERRORC25
		DNP1 = RK(I)/H	ERRORC26
		DN = QK(I)/HN	ERRORC27
		DNM1 = PK(I)/HNM1	ERRORC28
		FAC2 = DABS((DNP1 - DN) - C*(DN - DNM1))	ERRORC29
C		CHECK FOR CATASTROPHIC SUBTRACTION	ERRORC30
		IF (FAC2 .LE. ABS(DNP1)*1.0E-04) GO TO 5	ERRORC31
		E(I) = FAC1*(FAC2/ABSY)	ERRORC32
		5 IF (I .LE. 3 .OR. I3 .EQ. 0) GO TO 3	ERRORC33
		I1 = I - 3	ERRORC34
C		SKIP NEGLECTED SPECIES	ERRORC35
		DO 2 J=1,I1	ERRORC36
		IF (NEGL(J) .EQ. I1) GO TO 4	ERRORC37
		2 CONTINUE	ERRORC38
		3 IF (E(I) .LE. ERRORC) GO TO 4	ERRORC39
		ERRORC = F(I)	ERRORC40
		JC = J	ERRORC41
		4 CONTINUE	ERRORC42
C			ERRORC43
		RETURN	ERRORC44
		END	ERRORC45

*FILL		FILL0001
	SUBROUTINE FILL(X,Y,NA,NB)	FILL0002
CFILL		FILL0003
C	LINEAR INTERPOLATION TO FIL VACANCIES IN INPUT LISTS	FILL0004
	COMMON /CBITS/BITS	FILL0005
	DIMENSION X(10),Y(10)	FILL0006
C	FIND IA,IB = VACANT REGION	FILL0007
	IA=NA+1	FILL0008
	IF(Y(IA-1).EQ.BITS) GO TO 99	FILL0009
3	DO 4 I=IA,NB	FILL0010
	IF(Y(I).NE.BITS) GO TO 5	FILL0011
4	CONTINUE	FILL0012
	IB=NB	FILL0013
	GO TO 7	FILL0014
5	IB=I-1	FILL0015
	IF(I.EQ.IA) GO TO 12	FILL0016
C	FILL VACANCIES	FILL0017
	IF(Y(IB+1).NE.Y(IA-1)) GO TO 9	FILL0018
C	ALL VALUES THE SAME	FILL0019
7	DO 8 II=IA,IB	FILL0020
8	Y(II)=Y(IA-1)	FILL0021
	GO TO 12	FILL0022
C	INTERPOLATE	FILL0023
9	DX = X(IB+1) - X(IA-1)	FILL0024
	DO 11 II=IA,IB	FILL0025
11	Y(II) = (Y(IB+1)*(X(II)-X(IA-1)) + Y(IA-1)*(X(IB+1)-X(II)))/DX	FILL0026
C	GO BACK AND SEARCH FOR MORE REGIONS	FILL0027
12	IA = IB+2	FILL0028
	IF(I.LT.NB) GO TO 3	FILL0029
99	RETURN	FILL0030
	END	FILL0031

C FUEL THERMO PROPERTIES OF RAW FUEL VAPOR

C SUBROUTINE FUEL(N,TR,H,CP)

C REAL N,C(7,2)

C N IS H/C RATIO OF FUEL

C C ARE NASA COEFFICIENTS FOR C2H4

DATA T1D,TMID,TM1/300.0,1000.0,5000.0,
C/3.4552152,1.1491803E-2,-4.3651750E-6,7.6155095E-10,
-5.0123200E-14,6.7694946E3,2.6987959,
1.4256821,1.1383140E-2,7.9890006E-6,-1.6253679E-8,
16.7491256E-12,7.6292582E3,1.4621819E1/

C PNDM(T)=A1+T*(A2+T*(A3+T*(A4+T*A5)))
TK=TR/1.8

C IF(TK,GE,0.35*TLOW,AND,TK,LE,TM1)GO TO 20
WRITE(6,999)TR
IF(TK,GT,0.0,AND,TK,LE,1.20*TM1)GO TO 10
CALL ERROR
10 WRITE(6,998)

C 999 FORMAT(10H FUEL TEMP,F9.1,22H DEGR IS OUT OF RANGE.)
998 FORMAT(30H EXTRAPOLATED VALUES RETURNED.)

C SELECT TEMPERATURE RANGE
20 K=2

C IF(TK,GT,TMID)K=1

C COMPUTE ENTHALPY OF ETHYLENE

A1=C(1,K)+C(6,K)/TK
A2=C(2,K)/2.0
A3=C(3,K)/3.0
A4=C(4,K)/4.0
A5=C(5,K)/5.0
HRT=PNDM(TK)
H=HRT*1.98596*TR/28.054

C COMPUTE SPECIFIC HEAT

C A1=C(1,K)
A2=C(2,K)
A3=C(3,K)
A4=C(4,K)
A5=C(5,K)
CP=PNDM(TK)*1.98596/28.054

C

FUEL0001
FUEL0002
FUEL0003
FUEL0004
FUEL0005
FUEL0006
FUEL0007
FUEL0008
FUEL0009
FUEL0010
FUEL0011
FUEL0012
FUEL0013
FUEL0014
FUEL0015
FUEL0016
FUEL0017
FUEL0018
FUEL0019
FUEL0020
FUEL0021
FUEL0022
FUEL0023
FUEL0024
FUEL0025
FUEL0026
FUEL0027
FUEL0028
FUEL0029
FUEL0030
FUEL0031
FUEL0032
FUEL0033
FUEL0034
FUEL0035
FUEL0036
FUEL0037
FUEL0038
FUEL0039
FUEL0040
FUEL0041
FUEL0042
FUEL0043
FUEL0044
FUEL0045
FUEL0046
FUEL0047
FUEL0048
FUEL0049
FUEL0050

C GE STANDARD GUESS AT JET FUEL HEATING VALUE

FUEL0051

C $HV = (184686.04 + 37977.7 \cdot N) / (11.91468 + N)$

FUEL0052

FUEL0053

C FUEL0054

C HEAT OF VAPORIZATION AT 77F IS ABOUT 155 B/LB.

FUEL0055

C FUEL0056

H = H - 20275.0 + HV + 155.0

FUEL0057

RETURN

FUEL0058

END

FUEL0059

```

CGAMCP      COMPUTE GAMMA, CP=--- (P.W. HECK)
SURROUTINE GAMCP( TT,GAM,CP,RGAS,J1,J2)
DIMENSION TT(1),GAM(1),CP(1)

```

```

C*
C* VALID TEMPERATURE RANGE= T.LT.3600
C*

```

```

1 ROJ=RGAS/778.
DO 10 L=J1,J2
T=TT(L)
IF(T.LE.800.) GO TO 50
IF(T.GE.3600.) GO TO 40

```

```

C*
C* GAM(L)=2.23708/T**.070271
GO TO 10

```

```

C*
C* 50 GAM(L)=1.4
GO TO 10

```

```

C*
C* 40 GAM(L)=1.254

```

```

C*
C* 10 CP(L)=GAM(L)*ROJ/(GAM(L)-1.)
RETURN
END

```

```

GAMCP001
GAMCP002
GAMCP003
GAMCP004
GAMCP005
GAMCP006
GAMCP007
GAMCP008
GAMCP009
GAMCP010
GAMCP011
GAMCP012
GAMCP013
GAMCP014
GAMCP015
GAMCP016
GAMCP017
GAMCP018
GAMCP019
GAMCP020
GAMCP021
GAMCP022
GAMCP023

```

CGAMH FUNCTION GAMH (T)=P.H. HECK

FUNCTION GAMH(T)

IF(T,LE. 800.) GO TO 10

IF(T,GE. 3600.) GO TO 12

C*

GAMH=2.23768/T**0.070291

GO TO 15

10 GAMH=1.4

GO TO 15

12 GAMH=1.254

15 RETURN

END

GAMH0001

GAMH0002

GAMH0003

GAMH0004

GAMH0005

GAMH0006

GAMH0007

GAMH0008

GAMH0009

GAMH0010

GAMH0011

GAMH0012

CGCKP	MAIN DRIVER ROUTINE FOR GCKP-1 CALC.	GCKP0001
	SUBROUTINE GCKP(KK)	GCKP0002
	COMMON /TROUHL/ FRR,ERRMAJ,INERR,PRERR	GCKP0003
	LOGICAL ERR,ERRMAJ,INERR,PRERR	GCKP0004
	COMMON /INDATA/ N,HF,WAR,T2,BETA,T25,FARS,EINQ2C,PQ,	GCKP0005
*	RCO(11),RCO2(11),RHC(11),RNDX(11),	GCKP0006
*	PT(11),PS(11),BLOC(11),QCO(11)	GCKP0007
	REAL N	GCKP0008
	COMMON /JETDAT/ NPTS,RAD(12),TS(12),H(12),SPV(12),MWT(12),	GCKP0009
*	CP(12),FUUI(12),SPALDG(12),TKE(12),OTHER1(36)	GCKP0010
	REAL MWT	GCKP0011
	COMMON /GASCOMP/ RICH(12,2),FUEL(2,12,2),ENTH(2,12,2),	GCKP0012
*	CONC(16,2,12,2),HCINCP(12,2),OTHER2(192)	GCKP0013
	COMMON /GASTHW/ TG(2,12,2),MWTG(2,12,2),TAU(12,2),CPG(2,12,2)	GCKP0014
	REAL MWTG	GCKP0015
	COMMON /STCTRL/ LSTA,FINAL,CHEMK,FIRSTM,FIRSTC,XC,DXC,	GCKP0016
*	NIST,POUT1,DUMST1(8)	GCKP0017
	LOGICAL FINAL,FIRSTM,FIRSTC	GCKP0018
	LOGICAL POUT1	GCKP0019
	INTEGER CHEMK	GCKP0020
	COMMON /CPRINT/ PDUM(20)	GCKP0021
	COMMON /COND / DVAR,AREA,MODT,P,IVAR,V,RHO,T,C(25),LS,DUM2(2)	GCKP0022
	COMMON /PRIN / PRINT(50),NPRNTS,END,EVSTEP	GCKP0023
	REAL IVAR	GCKP0024
C		GCKP0025
	POUT1 = .FALSE.	GCKP0026
	K = KK	GCKP0027
C		GCKP0028
C	SFT GCKP COMMONS	GCKP0029
C		GCKP0030
	P = PQ*144.	GCKP0031
	T = TG(1,K,2)	GCKP0032
	IF(T.LE.1500.) GO TO 201	GCKP0033
	DVAR = 0.	GCKP0034
	IVAR = TAU(K,1)	GCKP0035
	NPRNTS = 1	GCKP0036
	PRINT(1) = TAU(K,2)	GCKP0037
	XMW = MWTG(1,K,2)	GCKP0038
	CALL FMPYC(1,XMW,CONC(1,1,K,2),C,16)	GCKP0039
	IF(FINAL .AND. LSTA.EQ.1 .AND. K.EQ.1) POUT1 = .TRUE.	GCKP0040
	IF(FINAL) WRITE (6,50) K	GCKP0041
	IF(PDUM(11).NE.0.) WRITE (6,50) K	GCKP0042
50	FORMAT(1H1,//////////47X,27H*** S T R E A M T	GCKP0043
	*H R F,2X,12,3X,3H***)	GCKP0044
	CALL GCKP1(\$199)	GCKP0045
	FIRSTC = .FALSE.	GCKP0046
	CALL MOVE(1,C,CONC(1,1,K,1),16,1)	GCKP0047
	TG(1,K,1) = T*1.6	GCKP0048
	GO TO 200	GCKP0049
199	FRR = .TRUE.	GCKP0050

200 RETURN

201 TG(1,K,1)= TG(1,K,2)

CALL MOVE(1,CONC(1,1,K,2),CONC(1,1,K,1),16,1)

GO TO 200

END

GCKP0051

GCKP0052

GCKP0053

GCKP0054

GCKP0055

CGCKP1	GENERAL CHEMICAL KINETICS PROGRAM	GCKP1001
C	SUBROUTINE GCKP1(*)	GCKP1002
C	OBTAINED FROM DA BITTKER OF NASA 11/29/71	GCKP1003
C		GCKP1004
C	LOGICAL NEXT,EVSTEP	GCKP1005
C		GCKP1006
C	REAL IVAR	GCKP1007
C		GCKP1008
C	COMMON/COND/DUM1(4),IVAR,Y(2A),DUM2(2),NEXT	GCKP1009
C	COMMON/SINT/HMIN,HINT,HN,HNP1,HMAX,NH,AVH,EMAX,ERRN,JCV,KOUNT,ERRP	GCKP1010
C	COMMON/PORF/PK(28),OK(28),KK(28),F(2A)	GCKP1011
C	COMMON/PRIN/PRINT(50),NPRNTS,END,EVSTEP	GCKP1012
C	COMMON /STCTRL/ ISTA,FINAL,CHEMK,FIRSTM,FIRSTC,XC,DXC,	GCKP1013
C	* NIST,POUT1,DUMST1(8)	GCKP1014
C	LOGICAL FIRSTC,FINAL	GCKP1015
C	LOGICAL POUT1	GCKP1016
C	COMMON /CPRINT/ PDUM(20)	GCKP1017
C		GCKP1018
C	READ AND CONVERT INPUT, PERFORM PRE-KINETIC CALCULATIONS	GCKP1019
C	CALL KINP	GCKP1020
C	1 IF (NEXT) GO TO 1000	GCKP1021
C		GCKP1022
C	PRINT REACTIONS, ASSIGNED VARIABLE PROFILE, INTEGRATION CONTROLS	GCKP1023
C		GCKP1024
C	IF(FINAL .AND. POUT1) CALL OUT1	GCKP1025
C		GCKP1026
C	COMPUTE (NON-INPUT) INITIAL CONDITIONS	GCKP1027
C	CALL PRED1	GCKP1028
C		GCKP1029
C	PRINT ALL INITIAL CONDITIONS	GCKP1030
C	IF(FINAL) CALL OUT2	GCKP1031
C	IF(PDUM(11).NE.0.) CALL OUT2	GCKP1032
C	IF (NEXT) GO TO 1000	GCKP1033
C		GCKP1034
C	INITIAL INTEGRATION STEPS	GCKP1035
C	CALL INT1	GCKP1036
C	NH = 2	GCKP1037
C	AVH = HINT	GCKP1038
C	IF (NEXT) GO TO 1000	GCKP1039
C		GCKP1040
C	IF (.NOT. EVSTEP) GO TO 3	GCKP1041
C		GCKP1042
C	** INTEGRATION - PRINT RESULTS AFTER EVERY STEP	GCKP1043
C	2 NH = 0	GCKP1044
C	PREV = IVAR	GCKP1045
C	CALL INTG	GCKP1046
C	NH = NH + 1	GCKP1047
C	AVH = HN	GCKP1048
C	IF (NH.NE.1) AVH = (IVAR - PREV)/FLOAT(NH)	GCKP1049
C	IF(FINAL) CALL OUT3	GCKP1050

IF (NEXT) GO TO 1000	GCKP1051
HTOP = END - IVAR	GCKP1052
IF (HTOP .IE. 0.) GO TO 100	GCKP1053
IF ((HTOP-HNP1) .LT. 0.) HNP1 = HTOP	GCKP1054
GO TO 2	GCKP1055
C	GCKP1056
C LOCATE FIRST PRINT STATION	GCKP1057
3 DO 4 I=1,NPRNTS	GCKP1058
IF (PRINT(I) .GT. IVAR) GO TO 5	GCKP1059
4 CONTINUE	GCKP1060
5 NS = I	GCKP1061
C	GCKP1062
C ** INTEGRATION - PRINT RESULTS AT PRINT STATIONS	GCKP1063
DO 10 I=NS,NPRNTS	GCKP1064
NH = 0	GCKP1065
PREV = IVAR	GCKP1066
C	GCKP1067
C SET NEXT PRINT STATION	GCKP1068
PRNT = PRINT(I)	GCKP1069
C INTEGRATE TO PRINT STATION	GCKP1070
6 HTOP = PRNT - IVAR	GCKP1071
IF (HTOP .IE. HNP1) GO TO 7	GCKP1072
CALL INTG	GCKP1073
NH = NH + 1	GCKP1074
IF (NEXT) GO TO 8	GCKP1075
GO TO 6	GCKP1076
C	GCKP1077
C SPECIAL STEP TO PRINT STATION	GCKP1078
7 CALL CASG (HN,QK,HTOP,RK)	GCKP1079
ERRN = ERRNRC(Y,RK,E,JCV,HTOP)	GCKP1080
CALL PRED	GCKP1081
NH = NH + 1	GCKP1082
8 AVH = (IVAR - PREV)/FLOAT(NH)	GCKP1083
IF (FINAL) CALL OUT3	GCKP1084
IF (PDUM(11).NE.0.) CALL OUT3	GCKP1085
IF (NEXT) GO TO 1000	GCKP1086
CALL INTG	GCKP1087
C	GCKP1088
10 CONTINUE	GCKP1089
C	GCKP1090
100 RETURN	GCKP1091
C	GCKP1092
1000 WRITE (6,1001)	GCKP1093
1001 FORMAT (7H0(GCKP),5X,46H A FATAL ERROR HAS OCCURRED - CASE TERMINATING	GCKP1094
*ATED)	GCKP1095
RETURN 1	GCKP1096
C	GCKP1097
END	GCKP1098

CGMDD	MODIFY INITIAL G PROFILES TO FORCE RAPID	GMDD0001
C	HOMOGENIZATION OF LOW YRICH'S	GMDD0002
C		GMDD0003
	SUBROUTINE GMDDEY	GMDD0004
	COMMON /CHITS / HITS,BLANK	GMDD0005
	REAL MJET,MUREF,N	GMDD0006
	DIMENSION CNAME(12),ALE(12),SCM(12),CPC(36),GJET(200),Y(200),	GMDD0007
	* UD(200),THD(200),TID(200),ED(200)	GMDD0008
	REAL MOLF1(100),MOLF2(100),MOLF3(100),MOLF4(100),	GMDD0009
	MOLF5(100),MOLF6(100),MOLF7(100),MOLF8(100),	GMDD0010
	MOLF9(100),MOLF10(100),MOLF11(100),MOLF12(100)	GMDD0011
	DIMENSION RAD(12),TS(12),UJ(12),SPV(12)	GMDD0012
	INTEGER MWT(12)	GMDD0013
	DIMENSION CP(12),FUEL(12),SPALDG(12),TKE(12),OTHER(36),	GMDD0014
	TITLE(20),PRINT(30),RCU(11),RCU2(11),RHC(11),RNOX(11),PT(11),	GMDD0015
	PS(11),BLJC(11),RCJ(11),RICH(24),HCINCP(24),FUUL(48),ENTH(48),	GMDD0016
	CONC(768),OTHER2(192)	GMDD0017
	DIMENSION SPLDGI(12),F(2,12,2),PDUM(20)	GMDD0018
	EQUIVALENCE (FUUL(1),F(1,1,1))	GMDD0019
C		GMDD0020
C		GMDD0021
	NAMELIST/A/YMIN,PDUM,YAIR	GMDD0022
	NAMELIST/GMDDOT/NPTS,YMIN,FUEL,F,SPALDG,SPLDGI,GJET,NJ,RICH	GMDD0023
C		GMDD0024
	DATA PDUM/20*0.0/	GMDD0025
C		GMDD0026
C		GMDD0027
C	READ PREJET INPUT FILE	GMDD0028
C		GMDD0029
	1 READ (1)	GMDD0030
	* DIAJ,MJET,TIJET,VJET,PE,TF,TIE,VE,GFX,RG,PR,PRT,SC,TREF,	GMDD0031
	* MUREF,DIFF,NC,CNAME,ALE,SCM,CPC,NJ,NM,CT1,CT2,CT3,CT4,CT5,	GMDD0032
	* CT6,CT7,CT8,GJET,Y,UD,THD,TID,ED,	GMDD0033
	* MOLF1,MOLF2,MOLF3,MOLF4,MOLF5,MOLF6,	GMDD0034
	* MOLF7,MOLF8,MOLF9,MOLF10,MOLF11,MOLF12,	GMDD0035
	* NPT, RAD, TS, UJ, SPV, MWI, CP, FUEL, SPALDG, TKE, OTHER,	GMDD0036
	* TITLE, PRINT, N, HF, WAR, T2, BETA, T25, FARS, FINO2C, PO,	GMDD0037
	* RCJ,RCJ2,RHC,RNOX,P1,PS,BLJC,RCO,	GMDD0038
	* RICH,HCINCP,FUUL,ENTH,CONC,OTHER2	GMDD0039
	* ,GEX	GMDD0040
	REWIND 1	GMDD0041
C		GMDD0042
	YMIN=0.30	GMDD0043
	YAIR=HITS	GMDD0044
	READ(5,A)	GMDD0045
C		GMDD0046
	YMIN1=(1.0-YMIN)/YMIN	GMDD0047
	DO 10 NN=1,NPTS	GMDD0048
	10 SPLDGI(NN)=AMINI(SPALDG(NN),YMIN1*(FUEL(NN)-F(2,NN,1))*2)	GMDD0049
	GJET(1)=SPLDGI(1)	GMDD0050

NPP1=NPTS+1	GMDD0051
DO 50 NN=1,NPP1	GMDD0052
J=6*(NN-1)	GMDD0053
IF(NN.EQ.1)GO TO 20	GMDD0054
GJET(J)=0.5*(SPLDG1(NN-1)+SPLDG1(NN))	GMDD0055
GJET(J+1)=0.1*SPLDG1(NN-1)+0.9*SPLDG1(NN)	GMDD0056
IF(NN.EQ.NPP1)GO TO 40	GMDD0057
20 JP2=J+2	GMDD0058
JP4=J+4	GMDD0059
DO 30 J=JP2,JP4	GMDD0060
30 GJET(J)=SPLDG1(NN)	GMDD0061
GJET(JP4+1)=0.9*SPLDG1(NN)+0.1*SPLDG1(NN+1)	GMDD0062
GO TO 50	GMDD0063
40 GJET(J+2)=SPLDG1(NN)	GMDD0064
50 CONTINUE	GMDD0065
C	GMDD0066
IF(YAIR.NE.BITS)RICH(NPP1)=YAIR	GMDD0067
IF(PDUM(1).NE.0.0)WRITE(6,GMDD0068)	GMDD0068
C	GMDD0069
C	GMDD0070
C REWRITE PREJET FILE	GMDD0071
C	GMDD0072
WRITE (1)	GMDD0073
* DIAJ,MJET,TIJET,VJET,PE,TF,TIF,VE,GFX,RG,PR,PRT,SC,TRFF,	GMDD0074
* MUREF,DIFF,NC,CNAME,ALE,SCM,CPC,NJ,NM,CT1,CT2,CT3,CT4,CT5,	GMDD0075
* CT6,CT7,CT8,GJET,Y,UD,THD,TID,ED,	GMDD0076
* MOLF1,MOLF2,MOLF3,MOLF4,MOLF5,MOLF6,	GMDD0077
* MOLF7,MOLF8,MOLF9,MOLF10,MOLF11,MOLF12,	GMDD0078
* NP1,RAJ,T3,UJ,SPV,MW1,CP,FUEL,SPALOG,TKF,OTHER,	GMDD0079
* TFILE,PRINT,N,HF,WAR,T2,HCTA,T25,FARS,EIND2C,PO,	GMDD0080
* RCD,RCDP,RHC,RNOX,P1,PS,BLOC,QCU,	GMDD0081
* RICH,HCINCP,FUUL,ENTH,CONC,OTHER2	GMDD0082
* ,GEX	GMDD0083
RETURN	GMDD0084
END	GMDD0085

CHCEQ2	FULL-EQUILIBRIUM COMPOSITION CH(N)-AIR-H2O SYSTEM	HCEQ2001
C	BRUTE-FORCE METHOD	HCEQ2002
C	LIKE HCEQ3 EXCEPT COMPOSITION RETURNED MOLE FRACTIONS	HCEQ2003
C		HCEQ2004
C	SUBROUTINE HCEQ2(FAR,WAR,HC,T,P, FIXCO, FIXNO, ZZ)	HCEQ2005
C		HCEQ2006
	LOGICAL FIXCO, FIXNO, SAMIX, TROUBL, NOSOLN, DONE	HCEQ2007
	LOGICAL DISQC	HCEQ2008
	REAL MF, K1, K2, K3, K4, K5, LNK6, LNP6	HCEQ2009
	DIMENSION Z(11), ZZ(11)	HCEQ2010
	DIMENSION B(3), A(9), DZ(3)	HCEQ2011
	COMMON /GHSC / FF(25), HH(25), SR(25), CPZ(25), DCPR(25)	HCEQ2012
C		HCEQ2013
	EQUIVALENCE(DSUMH, B(1)), (DSUMJ, B(2)), (DPK6, B(3)),	HCEQ2014
	(DZH2, DZ(1)), (DZ02, DZ(2)), (DZH2O, DZ(3))	HCEQ2015
C		HCEQ2016
C	ORDER OF SPECIES = H, O, H2, O2, OH, H2O, CO, CO2, N2, A, NO	HCEQ2017
C		HCEQ2018
	TROUBL = .FALSE.	HCEQ2019
	SAMIX = (FAR - JLD FAR + WAR - OLD WAR + HC - OLD HC), EQ, 0.0	HCEQ2020
	IF (SAMIX) GO TO 10	HCEQ2021
C		HCEQ2022
C	CALCULATE EQUIVALENCE RATIO	HCEQ2023
C		HCEQ2024
	OLD FAR = FAR	HCEQ2025
	OLD WAR = WAR	HCEQ2026
	OLD HC = HC	HCEQ2027
	MF = 12.01 + 1.008 * HC	HCEQ2028
	FARS = 0.209495 / 28.9666 * MF / (1.0 + 0.25 * HC)	HCEQ2029
	ER = FAR / FARS	HCEQ2030
C		HCEQ2031
C	TOTAL POUND-ATOMS EACH CONSTITUENT PER POUND MIX	HCEQ2032
C		HCEQ2033
	GAR = 1.0 + FAR + WAR	HCEQ2034
	SUMH = (FAR * HC / MF + WAR * 2.0 / 18.016) / GAR	HCEQ2035
	SUMC = (FAR / MF + 3E-4 / 28.9666) / GAR	HCEQ2036
	SUMJ = (WAR / 18.016 + 2.0 * (0.209495 + 3E-4) / 28.9666) / GAR	HCEQ2037
	SUMN = 2.0 * 0.780881 / 28.9666 / GAR	HCEQ2038
	SUMA = 0.009324 / 28.9666 / GAR	HCEQ2039
	CC = 2.0 * SUMC + 0.5 * SUMH - SUMJ	HCEQ2040
	ZZX = 0.5 * (3E-4 + 28.9666 / 18.016 * WAR) * SUMN / 0.780881	HCEQ2041
	CCL = SUMC + 0.5 * SUMH - ZZX	HCEQ2042
	IF (CCL, LE, 0.0) CCL = 1E-15	HCEQ2043
	CCR = 0.5 * (SUMJ - SUMC - ZZX)	HCEQ2044
10	BETA1 = 0.01	HCEQ2045
	DPK6 = 0.0	HCEQ2046
C		HCEQ2047
C	SET CONCENTRATIONS OF ISOLATED SPECIES, ZERO OTHERS,	HCEQ2048
C		HCEQ2049
	CALL SETM(1, 0, 0, Z, 11)	HCEQ2050

Z(10)=SUMA	HCEQ2051
IF(FIXCD)Z(7)=ZZ(7)/(ZZ(7)+ZZ(8))*SUMC	HCEQ2052
IF(FIXND)Z(11)=ZZ(11)/(ZZ(11)+2.0*ZZ(9))*SUMH	HCEQ2053
IF(T.EQ.OLDT)GO TO 20	HCEQ2054
C	HCEQ2055
C GET EQUILIBRIUM CONSTANTS	HCEQ2056
C	HCEQ2057
OLDT=T	HCEQ2058
TXK = T / 1.8	HCEQ2059
CALL THRM(TXK , 1.)	HCEQ2060
K1=EXP(FF(3)+FF(4)-2.0*FF(5))	HCEQ2061
K2=EXP(FF(3)+FF(4)-FF(6)-FF(2))	HCEQ2062
K3=FF(3)+FF(5)-FF(6)-FF(1)	HCEQ2063
IF(K3.GT.88.0)K3=88.0	HCEQ2064
K3=EXP(K3)	HCEQ2065
K4=EXP(FF(8)+FF(3)-FF(6)-FF(7))	HCEQ2066
K5=EXP(FF(9)+FF(4)-2.0*FF(11))	HCEQ2067
LNK6= FF(1)+FF(5)-FF(6)	HCEQ2068
20 LNP6=LNK6+ALOG(P/14.696)	HCEQ2069
DISSOC=(LNP6,LE,60.0).AND.(CCR,GT,1E+8)	HCEQ2070
IF(.NOT.DISSOC)BETA1=0.0	HCEQ2071
C	HCEQ2072
C INITIALIZE PRIMARY SPECIES	HCEQ2073
C	HCEQ2074
IF(ER,GT,1.0)GO TO 30	HCEQ2075
BETA1 = AMAX1(BETA1, Z(7)/CCL, (Z(11)+CC)/CCL)	HCEQ2076
BETA1 = BETA1*1.0000001	HCEQ2077
Z(3)=BETA1*CCL-Z(7)	HCEQ2078
Z(4)=0.5*(Z(3)+Z(7)-Z(11)-CC)	HCEQ2079
GO TO 40	HCEQ2080
30 BETA1 = AMAX1(BETA1, 0.5*Z(11)/CCR, 0.5*(Z(7)-CC)/CCR)	HCEQ2081
BETA1 = BETA1*1.0000001	HCEQ2082
Z(4)=BETA1*CCR-0.5*Z(11)	HCEQ2083
IF(FIXCD)GO TO 35	HCEQ2084
AAA=K4-1.0	HCEQ2085
CCC=CC+2.0*Z(4)+Z(11)	HCEQ2086
BBB=-(CCC*AAA+K4*SUMC+0.5*SUMH)	HCEQ2087
CCC=CCC*K4*SUMC	HCEQ2088
Z(7)=-0.5*(BBB+SQRT(BBB**2-4.0*AAA*CCC))/AAA	HCEQ2089
35 Z(3)=CC+2.0*Z(4)-Z(7)+Z(11)	HCEQ2090
40 Z(6)=SUMD+Z(7)-Z(11)-2.0*(SUMC+Z(4))	HCEQ2091
IF(DISSOC.AND.(Z(4),LE,0.0)) Z(4)=1E-30	HCEQ2092
C	HCEQ2093
C BEGIN ITERATION	HCEQ2094
C	HCEQ2095
DO 100 I=1,30	HCEQ2096
C	HCEQ2097
C RADICALS FROM EQUILIBRIA	HCEQ2098
C	HCEQ2099
Z(5)=SQRT(K1*Z(3)*Z(4))	HCEQ2100

	Z(2)=K2*Z(3)*Z(4)/Z(6)	HCEQ2101
	Z(1)=K3*Z(3)*Z(5)/Z(6)	HCEQ2102
	IF(FIXCD)GO TO 50	HCEQ2103
	Z(7)=K4*SUMC*Z(3)/(Z(6)+K4*Z(3))	HCEQ2104
50	Z(8)=SUMC-Z(7)	HCEQ2105
	IF(FIXND)GO TO 60	HCEQ2106
	ZD2K5=0.0	HCEQ2107
	IF(DISSDC)ZD2K5=0.25*K5*Z(4)	HCEQ2108
	Z(11)=SQRT(ZD2K5*(ZD2K5+2.0*SUNN))-ZD2K5	HCEQ2109
60	Z(9)=0.5*(SUMN-Z(11))	HCEQ2110
C		HCEQ2111
C	H ₂ O CONTINUITY DEFECTS AND 3-BODY EQUILIBRIUM DEFECT	HCEQ2112
C		HCEQ2113
	DSUMH=SUMH-Z(1)-Z(5)-2.0*(Z(3)+Z(6))	HCEQ2114
	DSUMD=SUMD-Z(2)-Z(5)-Z(6)-Z(7)-Z(11)-2.0*(Z(4)+Z(8))	HCEQ2115
	SPM=0.0	HCEQ2116
	DO 70 J=1,11	HCEQ2117
70	SPM=SPM+Z(J)	HCEQ2118
	IF(DISSDC)DPK6=LNPK6-ALOG(Z(6)/Z(1)*SPM/Z(5))	HCEQ2119
C		HCEQ2120
C	TEST FOR CONVERGENCE	HCEQ2121
C		HCEQ2122
	DONE=(ABS(DSUMH).LT.1E-6*SUMH)	HCEQ2123
	AND.(ABS(DSUMD).LT.1E-6*SUMD)	HCEQ2124
	AND.(ABS(DPK6).LT.1E-6)	HCEQ2125
	IF(DONE)GO TO 99	HCEQ2126
C		HCEQ2127
C	PARTIAL DERIVATIVES TO DIRECT CONVERGENCE	HCEQ2128
C		HCEQ2129
C		HCEQ2130
C	UNDISSOCIATED RICH MIXTURES	HCEQ2131
C		HCEQ2132
	IF(DISSDC)GO TO 75	HCEQ2133
	ZCDH2=Z(7)/Z(3)	HCEQ2134
	ZCDC=Z(7)/SUMC	HCEQ2135
	DZH2D=(DSUMJ)+0.5*ZCDH2*(1.0-ZCDC)*DSUMH	HCEQ2136
	/(1.0+ZCDH2+ZCDH2*ZCDC*(1.0/K4-1.0))	HCEQ2137
	DZH2=0.5*DSUMH-DZH2D	HCEQ2138
	DZD2=0.0	HCEQ2139
	GO TO 99	HCEQ2140
C		HCEQ2141
C		HCEQ2142
C	DISSOCIATED MIXTURES	HCEQ2143
C		HCEQ2144
75	Z3=AMAX1(Z(3),1E-36)	HCEQ2145
	Z4=AMAX1(Z(4),1E-36)	HCEQ2146
	A(1)=2.0+0.5*(3.0*Z(1)+Z(5))/Z3	HCEQ2147
	A(2)=(Z(2)+0.5*Z(5))/Z3	HCEQ2148
	DCODX=K4*SUMC/(Z(6)+K4*Z(3))*2	HCEQ2149
	IF(.NOT.FIXCD)A(2)=A(2)-DCODX*Z(6)	HCEQ2150

A(3)=(1.5*Z(1)+Z(2)+Z(3)+0.5*Z(5)-2.0*SPH)/SPH/Z3	HCEQ2151
A(4)=0.5*(Z(1)+Z(5))/Z4	HCEQ2152
A(5)=2.0*(Z(2)+0.5*Z(5))/Z4	HCEQ2153
DNOD02=0.0	HCEQ2154
IF(Z(4).GT.4E-38/K5)DNOD02=0.5*K5*Z(9)/(0.25*K5*Z(4)+Z(11))	HCEQ2155
IF(.NOT.FIXNO)A(5)=A(5)+DNOD02	HCEQ2156
A(6)=0.5*(Z(1)+Z(5))+Z(2)+Z(4)-SPH	HCEQ2157
IF(.NOT.FIXNO)A(6)=A(6)+0.5*DNOD02*Z(4)	HCEQ2158
A(6)=A(6)/SPH/Z4	HCEQ2159
A(7)=2.0-Z(1)/Z(6)	HCEQ2160
A(8)=1.0-Z(2)/Z(6)	HCEQ2161
IF(.NOT.FIXCO)A(8)=A(8)+DCNDX*Z(3)	HCEQ2162
A(9)=(Z(1)-Z(2)+Z(6)+2.0*SPH)/SPH/Z(6)	HCEQ2163
C	HCEQ2164
C NEW ESTIMATES OF PRIMARY SPECIES	HCEQ2165
C	HCEQ2166
CALL SOLV3(A,8,DZ,NOSOLN)	HCEQ2167
99 TROUBL= TROUBL.OR,NOSOLN.OR, (I,EQ,20)	HCEQ2168
C	HCEQ2169
C PROBLEM DIAGNOSTICS	HCEQ2170
C	HCEQ2171
IF(TROUBL)WRITE(6,1000)1,Z	HCEQ2172
1000 FORMAT(1X,I2,8X, 1P1E11,4)	HCEQ2173
IF(DONE)GO TO 110	HCEQ2174
C	HCEQ2175
DZLIM=-0.9*Z(3)	HCEQ2176
IF(DZH2.LT.DZLIM)DZH2=DZLIM	HCEQ2177
Z(3)=Z(3)+DZH2	HCEQ2178
DZLIM=-0.9*Z(4)	HCEQ2179
IF(DZ02.LT.DZLIM)DZ02=DZLIM	HCEQ2180
Z(4)=Z(4)+DZ02	HCEQ2181
Z(6)=Z(6)+DZH20	HCEQ2182
C	HCEQ2183
100 CONTINUE	HCEQ2184
TROUBL= .TRUE,	HCEQ2185
C	HCEQ2186
C PROBLEM DIAGNOSTICS	HCEQ2187
C	HCEQ2188
110 IF(TROUBL)WRITE(6,1001)FAR,WAR,HC,T, P ,FIXCO,FIXNO	HCEQ2189
1001 FORMAT(15H0 I H,	HCEQ2190
10X,1H0,10X,2HH2,9X,2H02,9X,2H0H,9X,3HH20,8X,	HCEQ2191
12H02,9X,3HC02,8X,2HN2,9X,2HAR,9X,2HND/	HCEQ2192
125HOPSEQ2 DIAGNOSTICS FAR=F9,6,6H WAR=F9,6,5H HC=E10,3,	HCEQ2193
14H T=F9,2,7H P=F10,5,8H FIXCO=L2,8H FIXNO=L2)	HCEQ2194
C	HCEQ2195
C COMPOSITION RETURNED AS MOLE FRACTIONS	HCEQ2196
C	HCEQ2197
CALL FMPYC(1,1.0/SPH,Z,ZZ,11)	HCEQ2198
C	HCEQ2199
RETURN	HCEQ2200

END

HCEQ2201

CHCEQ3	FULL-EQUILIBRIUM COMPOSITION CH(N)-AIR-H ₂ O SYSTEM	HCEQ3001
C	LIKE HCEQ2 EXCEPT COMPOSITION RETURNED MOLES/LB MIX	HCEQ3002
C		HCEQ3003
C	SUBROUTINE HCEQ2(FAR,WAR,HC,T,P, FIXCO, FIXNO, ZZ)	HCEQ3004
C		HCEQ3005
	LOGICAL FIXCO, FIXNO, SAMIX, TROUBL, NOSOLN, DONE	HCEQ3006
	LOGICAL DISOC	HCEQ3007
	REAL MF, K1, K2, K3, K4, K5, LNK6, LNP6	HCEQ3008
	DIMENSION Z(11), ZZ(11)	HCEQ3009
	DIMENSION B(3), A(9), DZ(3)	HCEQ3010
	COMMON /GHSC / FF(25), HH(25), SR(25), CPZ(25), DCPR(25)	HCEQ3011
C		HCEQ3012
	EQUIVAJ=ENCF(DSUMH, B(1)), (DSUMJ, B(2)), (DPK6, B(3)),	HCEQ3013
	(DZH2, DZ(1)), (DZ02, DZ(2)), (DZH20, DZ(3))	HCEQ3014
C		HCEQ3015
C	ORDER OF SPECIES = H, O, H2, O2, OH, H2O, CO, CO2, N2, A, NO	HCEQ3016
C		HCEQ3017
	TRoubl = .FALSE.	HCEQ3018
	SAMIX=(FAR-OLDFAR+WAR-OLDWAR+HC-OLDHC).EQ.0.0	HCEQ3019
	IF(SAMIX)GO TO 10	HCEQ3020
C		HCEQ3021
C	CALCULATE EQUIVALENCE RATIO	HCEQ3022
C		HCEQ3023
	OLDFAR=FAR	HCEQ3024
	OLDWAR=WAR	HCEQ3025
	OLDHC=HC	HCEQ3026
	MF=17.01+1.008*HC	HCEQ3027
	FARS=0.209495/28.9666*MF/(1.0+0.25*HC)	HCEQ3028
	ER=FAR/FARS	HCEQ3029
C		HCEQ3030
C	TOTAL POUND-ATOMS EACH CONSTITUENT PFR POUND MIX	HCEQ3031
C		HCEQ3032
	GAR=1.0+FAR+WAR	HCEQ3033
	SUMH=(FAR*HC/MF+WAR*2.0/18.016)/GAR	HCEQ3034
	SUMC=(FAR/MF+3E-4/28.9666)/GAR	HCEQ3035
	SUMJ=(WAR/18.016+2.0*(0.209495+3E-4)/28.9666)/GAR	HCEQ3036
	SUMN=2.0*0.780881/28.9666/GAR	HCEQ3037
	SUMA=0.009324/28.9666/GAR	HCEQ3038
	CC=2.0*SUMC+0.5*SUMH-SUMJ	HCEQ3039
	ZZX=0.5*(3E-4+28.9666/18.016*WAR)*SUMN/0.780881	HCEQ3040
	CCL=SUMC+0.5*SUMH-ZZX	HCEQ3041
	IF(CCL.LE.0.0)CCL=1E-15	HCEQ3042
	CCR=0.5*(SUMJ-SUMC-ZZX)	HCEQ3043
	10 RFTA1=0.01	HCEQ3044
	DPK6=0.0	HCEQ3045
C		HCEQ3046
C	SET CONCENTRATIONS OF ISOLATED SPECIES, ZERO OTHERS.	HCEQ3047
C		HCEQ3048
	CALL SFTM(1,0,0,7,11)	HCEQ3049
	Z(10)=SUMA	HCEQ3050

	IF (FIXC) Z(7) = ZZ(7) / (ZZ(7) + ZZ(8)) * SUMC	HCEQ3051
	IF (FIXN) Z(11) = ZZ(11) / (ZZ(11) + 2.0 * ZZ(9)) * SUMN	HCEQ3052
	IF (1.00, 0.01) GO TO 20	HCEQ3053
C	GET EQUILIBRIUM CONSTANTS	HCEQ3054
C		HCEQ3055
	MLDT = T	HCEQ3056
	TXK = T / 1.8	HCEQ3057
	CALL THRM (TXK, 1.)	HCEQ3058
	K1 = EXP (FF(3) + FF(4) - 2.0 * FF(5))	HCEQ3059
	K2 = EXP (FF(3) + FF(4) - FF(6) - FF(7))	HCEQ3060
	K3 = -FF(3) + FF(5) - FF(6) - FF(7)	HCEQ3061
	IF (K3, 0.0, 0.0) K3 = 88.0	HCEQ3062
	K3 = EXP (K3)	HCEQ3063
	K4 = EXP (FF(8) + FF(3) - FF(6) - FF(7))	HCEQ3064
	K5 = EXP (FF(9) + FF(4) - 2.0 * FF(11))	HCEQ3065
	LNK6 = FF(1) + FF(5) - FF(6)	HCEQ3066
20	LNPK6 = LNK6 + ALOG (P / 14.696)	HCEQ3067
	DISSOC = (LNPK6, LE, 60.0) .AND. (CCR, GT, 1E-8)	HCEQ3068
	IF (, NOT, DISSOC) BETA1 = 0.0	HCEQ3069
C		HCEQ3070
C	INITIALIZE PRIMARY SPECIES	HCEQ3071
C		HCEQ3072
	IF (CR, GT, 1.0) GO TO 30	HCEQ3073
	BETA1 = AMAX1 (BETA1, Z(7) / CCL, (Z(11) + CC) / CCL)	HCEQ3074
	BETA1 = BETA1 + 1.0000001	HCEQ3075
	Z(3) = BETA1 * CCL - Z(7)	HCEQ3076
	Z(4) = 0.5 * (Z(3) + Z(7) - Z(11) - CC)	HCEQ3077
	GO TO 40	HCEQ3078
30	BETA1 = AMAX1 (BETA1, 0.5 * Z(11) / CCR, 0.5 * (Z(7) - CC) / CCR)	HCEQ3079
	BETA1 = BETA1 + 1.0000001	HCEQ3080
	Z(4) = BETA1 * CCR - 0.5 * Z(11)	HCEQ3081
	IF (FIXC) GO TO 35	HCEQ3082
	AAA = K4 - 1.0	HCEQ3083
	CCC = CC + 2.0 * Z(4) + Z(11)	HCEQ3084
	BBB = - (CCC * AAA + K4 * SUMC + 0.5 * SUMN)	HCEQ3085
	CCC = CCC * K4 * SUMC	HCEQ3086
	Z(7) = -0.5 * (BBB + SQRT (BBB * 2 - 4.0 * AAA * CCC)) / AAA	HCEQ3087
35	Z(3) = CC + 2.0 * Z(4) - Z(7) + Z(11)	HCEQ3088
40	Z(6) = SUM7 + Z(7) - Z(11) - 2.0 * (SUMC + Z(4))	HCEQ3089
	IF (DISSOC, AND, (Z(4), LE, 0.0)) Z(4) = 1E-30	HCEQ3090
C		HCEQ3091
C	BEGIN ITERATION	HCEQ3092
C		HCEQ3093
	DO 100 I = 1, 30	HCEQ3094
C		HCEQ3095
C	RADICALS FROM EQUILIBRIA	HCEQ3096
C		HCEQ3097
	Z(5) = SQRT (K1 * Z(3) * Z(4))	HCEQ3098
	Z(2) = 2 * Z(3) * Z(4) / Z(6)	HCEQ3099
		HCEQ3100

	Z(1)=K3*7(3)*Z(5)/Z(6)	HCE03101
	IF(FIXC0)GO TO 50	HCE03102
	Z(7)=K4*SUMC*7(3)/(Z(6)+K4*Z(3))	HCE03103
50	7(A)=SUMC-7(7)	HCE03104
	IF(FIXN0)GO TO 60	HCE03105
	Z02K5=0.0	HCE03106
	IF(DISSOC)Z02K5=0.25*K5*Z(4)	HCE03107
	Z(11)=SQRT(Z02K5*(Z02K5+2.0*SUMN))-Z02K5	HCE03108
60	Z(9)=0.5*(SUMN-Z(11))	HCE03109
C		HCE03110
C	H ₂ O CONTINUITY DEFECTS AND 3-BODY EQUILIBRIUM DEFECT	HCE03111
C		HCE03112
	DSUMH=SUMH-Z(1)-Z(5)-2.0*(Z(3)+Z(6))	HCE03113
	DSUMN=SUMN-Z(2)-Z(5)-Z(6)-Z(7)-Z(11)-2.0*(Z(4)+Z(8))	HCE03114
	SPM=0.0	HCE03115
	DO 70 J=1,11	HCE03116
70	SPM=SPM+7(J)	HCE03117
	IF(DISSOC)DPK6=LDPK6-ALOG(7(6)/Z(1)*SPM/Z(5))	HCE03118
C		HCE03119
C	TEST FOR CONVERGENCE	HCE03120
C		HCE03121
	DONE=(ABS(DSUMH).LT.1E-6*SUMH)	HCE03122
	1.AND.(ABS(DSUMN).LT.1E-6*SUMN)	HCE03123
	1.AND.(ABS(DPK6).LT.1E-6)	HCE03124
	IF(DONE)GO TO 99	HCE03125
C		HCE03126
C	PARTIAL DERIVATIVES TO DIRECT CONVERGENCE	HCE03127
C		HCE03128
C	UNDISSOCIATED RICH MIXTURES	HCE03129
C		HCE03130
	IF(DISSOC)GO TO 75	HCE03131
	ZCNDH2=Z(7)/Z(3)	HCE03132
	ZCNDH2=7(7)/SUMC	HCE03133
	DZH20=(DSUMN+0.5*ZCNDH2*(1.0-ZCNDH2)*DSUMH)	HCE03134
	1/(1.0+ZCNDH2+ZCNDH2*ZCNDH2*(1.0/K4-1.0))	HCE03135
	DZH2=0.5*DSUMH-DZH20	HCE03136
	DZ02=0.0	HCE03137
	GO TO 99	HCE03138
C		HCE03139
C	DISSOCIATED MIXTURES	HCE03140
C		HCE03141
75	Z3=AMAX1(Z(3),1E-36)	HCE03142
	Z4=AMAX1(Z(4),1E-36)	HCE03143
	A(1)=2.0+0.5*(3.0*Z(1)+Z(5))/Z3	HCE03144
	A(2)=(7(2)+0.5*Z(5))/Z3	HCE03145
	DCNDX=K4*SUMC/(Z(6)+K4*Z(3))*2	HCE03146
	IF(.NOT.FIXC0)A(2)=A(2)-DCNDX*Z(6)	HCE03147
	A(3)=(1.5*7(1)+Z(2)+Z(3)+0.5*Z(5)-2.0*SPM)/SPM/Z3	HCE03148
	A(4)=0.5*(7(1)+Z(5))/Z4	HCE03149
	A(5)=2.0*(7(2)+0.5*Z(5))/Z4	HCE03150

DNODD2=0.0	HCF03151
IF(7(4).GT.4E-38/K5)DNODD2=0.5*K5*Z(9)/(0.25*K5*Z(4)+Z(1))	HCF03152
IF(.NOT.FIXNO)A(5)=A(5)+DNODD2	HCF03153
A(6)=0.5*(7(1)+7(5))+7(2)+Z(4)-SPM	HCF03154
IF(.NOT.FIXNO)A(6)=A(6)+0.5*DNODD2*Z(4)	HCF03155
A(6)=A(6)/SPM/Z4	HCF03156
A(7)=2.0-Z(1)/Z(6)	HCF03157
A(8)=1.0-Z(2)/Z(6)	HCF03158
IF(.NOT.FIXNO)A(8)=A(8)+DNODX*Z(3)	HCF03159
A(9)=(-Z(1)-Z(2)+Z(6)+2.0*SPM)/SPM/Z(6)	HCF03160
C	HCF03161
C NEW ESTIMATES OF PRIMARY SPECIES	HCF03162
C	HCF03163
CALL SOLV3(A,B,DZ,NDSOLN)	HCF03164
99 TROUBL= TROUBL.OR.NDSOLN.OR.(1.E0.20)	HCF03165
C	HCF03166
C PROBLEM DIAGNOSTICS	HCF03167
C	HCF03168
IF(TROUBL)WRITE(6,1000)I,Z	HCF03169
1000 FORMAT(IX,T2,8X,1P1E11.4)	HCF03170
IF(DNDF)GO TO 110	HCF03171
C	HCF03172
DZLIM=-0.99*Z(3)	HCF03173
IF(DZH2.LT.DZLIM)DZH2=DZLIM	HCF03174
Z(3)=Z(3)+DZH2	HCF03175
DZLIM=-0.99*Z(4)	HCF03176
IF(DZH2.LT.DZLIM)DZH2=DZLIM	HCF03177
Z(4)=Z(4)+DZH2	HCF03178
Z(6)=Z(6)+DZH20	HCF03179
C	HCF03180
100 CONTINUE	HCF03181
C	HCF03182
C PROBLEM DIAGNOSTICS	HCF03183
C	HCF03184
110 IF(TROUBL)WRITE(6,1001)FAR,WAR,HC,T,P,FXCO,FXNO	HCF03185
1001 FORMAT(15H1,10X,1H0,10X,2HH2,9X,2H02,9X,2HH0,9X,3HH20,8X,	HCF03186
12HC1,9X,3HC2,8X,2HH2,9X,2HAR,9X,2HNO,	HCF03187
125HOPSE02 DIAGNOSTICS FAR=F9.6,6H WAR=F9.6,5H HC=F10.3,	HCF03188
14H T=F9.2,7H P=F10.5,8H FXCO=12,8H FXNO=12)	HCF03189
C	HCF03190
C COMPOSITION RETURNED AS MOLES/LB MIX	HCF03191
C	HCF03192
CALL MOVE(1,Z,77,11,1)	HCF03193
C	HCF03194
RETURN	HCF03195
END	HCF03196
	HCF03197

CHYCARB	IGNITION DELAY FOR RAW FUEL	HYCARB01
	SUBROUTINE HYCARB(K)	HYCARB02
	COMMON /INDATA/ N,HF,WAR,T2,BETA,T25,FAR5,FINO2C,P0,	HYCARB03
*	RCO(1),RCO2(1),RHC(1),RNOX(1),	HYCARB04
*	PT(1),PS(1),BLOC(1),QCO(1)	HYCARB05
REAL	N	HYCARB06
COMMON /JETDAT/	NPTS,DUM(108),TFR(12),TFL,TIM1(12),TIM2,NTHER1(10)	HYCARB07
COMMON /GASCMP/	RICH(12,2),FUEL(2,12,2),ENTH(2,12,2),	HYCARB08
*	CONC(16,2,12,2),HCINCP(12,2),OTHER2(192)	HYCARB09
COMMON /GASTHW/	TG(2,12,2),MWTG(2,12,2),TAU(12,2),CPG(2,12,2)	HYCARB10
REAL	MWTG	HYCARB11
COMMON /OPCTRL/	TITLE(20),PRINT(30)	HYCARB12
COMMON /STCTRL/	ISTA,FINAL,CHEMK,FIRSTM,FIRSTC,XC,DXC,DUMST(10)	HYCARB13
LOGICAL	FINAL,FIRSTM,FIRSTC	HYCARB14
INTEGER	CHEMK	HYCARB15
COMMON /CPRINT/	PDUM(20)	HYCARB16
	DATA A,B/-17.671439,40290.880/	HYCARB17
C		HYCARB18
C		HYCARB19
C	CALCULATE IGNITION DELAY	HYCARB20
C		HYCARB21
	1 DTIME = TIM2-TIM1(K)	HYCARB22
	TEMP = TG(1,K,1)	HYCARB23
	DELAYI=1.E+30	HYCARB24
	IF(TEMP,GF,400.) DELAYI=.001*EXP(A,B/TEMP)	HYCARB25
C		HYCARB26
C	WEIGHT INITIAL INCIPIENCIES	HYCARB27
C		HYCARB28
	HCI = 0.	HYCARB29
	TOTF = 0.	HYCARB30
	DO 5 KK=1,NPTS	HYCARB31
	TOTF = TOTF+TFR(KK)	HYCARB32
	IF(HCINCP(KK,1).EQ. 1.) GO TO 5	HYCARB33
	HCI = HCI+TFR(KK)*HCINCP(KK,1)**4	HYCARB34
	5 CONTINUE	HYCARB35
	TOTF = TOTF+TFL	HYCARB36
	IF(PDUM(13).NE.0.) CALL TABPRI(6HIFRET,IFR,26,10)	HYCARB37
	IF(TOTF.LE.0.) GO TO 6	HYCARB38
	HCI = (HCI/TOTF)**.25	HYCARB39
	6 HCINCP(K,2)= HCI+DTIME/DELAYI	HYCARB40
	IF(PDUM(13).NE.0.) WRITE (6,11) K,HCI,DELAYI,TEMP,HCINCP(K,2)	HYCARB41
	11 FORMAT(2X,I3,3X,4E16.8/)	HYCARB42
	IF(HCINCP(K,2).LE.1.) GO TO 200	HYCARB43
C		HYCARB44
C	ADJUST CONC AND MW OF RICH FRACTION	HYCARB45
C		HYCARB46
C	REACTION== C10H10N + 5H2 = 10CO + (5N)H2	HYCARB47
C		HYCARB48
	C16 = CONC(16,1,K,2)	HYCARB49
	CONC(16,1,K,2)= 0.	HYCARB50

```
CONC(3,1,K,2)= CONC(3,1,K,2)+5.0*N*CI6  
CONC(7,1,K,2)= CONC(7,1,K,2)+10.0*CI6  
CONC(4,1,K,2)= CONC(4,1,K,2)-5.0*CI6  
HCINCP(K,2)= 1.  
200 RETURN  
END
```

```
HYCARN51  
HYCARH52  
HYCARB53  
HYCARB54  
HYCARB55  
HYCARB56
```

C	INTEE	SET UP AND CALL FOR INTEGRATION	INTEE001
		SUBROUTINE INTE	INTEE002
C			INTEE003
C		SET UP AND CALL FOR INTEGRATION	INTEE004
C			INTEE005
C		OBTAIN SUGGESTED STEP SIZE FOR NEXT INTEGRATION STEP	INTEE006
C			INTEE007
		EXTERNAL PPRR	INTEE008
C			INTEE009
C		LOGICAL NEXT,ELIM	INTEE010
C			INTEE011
C		REAL IVAR	INTEE012
C			INTEE013
C		DIMENSION YN(33),Y(28),C(25)	INTEE014
C			INTEE015
		COMMON/OPTS/VERST,TIMEV,VFSA,AREAV,FLIM,TCON,RHOCON,IPRCD	INTEE016
		COMMON/COND/YNP1(33),LS,SP3,NEXT	INTEE017
		COMMON/SINT/HMIN,HNM1,HN,HNP1,HMAX,NH,AVH,FMAX,ERRN,JCV,KOUNT,ERRP	INTEE018
		COMMON/PDRF/PK(28),OK(28),RK(28),E(28)	INTEE019
		COMMON/SPEC/SHAW(2,30),MW(25),W(25),STOIC(25,30),OMEGA(25,30)	INTEE020
C			INTEE021
		EQUIVALENC (IVAR,YNP1(5)),(Y,YNP1(6)),(C,YNP1(9)),(HINT,HNM1)	INTEE022
C			INTEE023
C		INITIAL STEPS OR RESTART	INTEE024
		ENTRY INTI	INTEE025
		NRFST = 0	INTEE026
		FMAX2 = FMAX/2	INTEE027
		FMAX56 = 5.*FMAX/6	INTEE028
		LSPB = LS + 8	INTEE029
		HNP1 = HINT	INTEE030
		DO 3 J=1,LSPB	INTEE031
		3 YN(I) = YNP1(I)	INTEE032
		2 CALL CAST (HN,OK,HNP1,PX)	INTEE033
		HN = HNP1	INTEE034
		CALL PRED	INTEE035
		CALL CAST (HN,PX,HNP1,OK)	INTEE036
		HNM1 = HN	INTEE037
		HN = HNP1	INTEE038
		CALL PRED	INTEE039
		DO 23 I=1,IS	INTEE040
		IF (C(I).GE.0.) GO TO 23	INTEE041
		WRITE (6,100)	INTEE042
		100 FORMAT (7H0(INTE),5X,43HCOMPOSITION ERROR = NEGATIVE CONCENTRATION	INTEE043
		*9)	INTEE044
		NEXT = .TRUE.	INTEE045
		RETURN	INTEE046
		23 CONTINUE	INTEE047
		RETURN	INTEE048
C			INTEE049
C		PREPARE TO CONTINUE INTEGRATION	INTEE050

ENTRY INTG	INTEE051
DO 4 I=1,LSP8	INTEE052
4 YNP1(I) = YN(I)	INTEE053
CALL PRED	INTEE054
RETURN	INTEE055
C	INTEE056
C GENERAL STEP	INTEE057
ENTRY INTG	INTEE058
CALL CASG (HN,QK,HNP1,RK)	INTEE059
C RELATIVE ERROR IN INTEGRATION STEP	INTEE060
ERRN = ERRORC(Y,RK,F,JCV,HNP1)	INTEE061
C TEST FOR RESTART CONDITIONS	INTEE062
IF (ERRN .IE. EMAX .OR. HNP1 .GT. HMIN) GO TO 7	INTEE063
C	INTEE064
WRITE (6,101) TVAR,(SNAM(J,JCV),J=1,2),HNM1,HN,HNP1	INTEE065
101 FORMAT (7H0(INTE),5X,7HRESTART,10X,2HINDEPENDENT VARIABLE =,E13.5	INTEE066
*,2X,11H(CGS UNITS),5X,2HCONTROLLING VARIABLE = ,2A4//51X,6HH(N-1)	INTEE067
*,17X,4HH(N),17X,6HH(N+1)/48X,3(E12.5,10X)//51X,6HK(N-1),17X,4HK(N)	INTEE068
*,17X,6HK(N+1),16X,6HE(N+1))	INTEE069
DO 5 I=1,LSP8	INTEE070
IF (I .GT. LSP3) GO TO 5	INTEE071
WRITE (6,102) (SNAM(J,I),J=1,2),PK(I),QK(I),RK(I),F(I)	INTEE072
102 FORMAT (30X,2A4,4(10X,E12.5))	INTEE073
5 YNP1(I) = YN(I)	INTEE074
NH = NH + 2	INTEE075
C	INTEE076
C STOP AFTER 10 RESTARTS	INTEE077
IF (NREST .LT. 10) GO TO 6	INTEE078
WRITE (6,103)	INTEE079
103 FORMAT (7H0(INTE),5X,25H10 RESTARTS HAVE OCCURRED)	INTEE080
NEXT = .TRUE.	INTEE081
RETURN	INTEE082
6 NREST = NREST + 1	INTEE083
HMIN = HMIN/2.	INTEE084
HNP1 = HMIN	INTEE085
GO TO 2	INTEE086
C	INTEE087
7 HNM1 = HN	INTEE088
HN = HNP1	INTEE089
DO 8 I=1,LSPA	INTEE090
IF (I .GT. LSP3) GO TO 8	INTEE091
PK(I) = QK(I)	INTEE092
QK(I) = RK(I)	INTEE093
8 YN(I) = YNP1(I)	INTEE094
CALL PRED	INTEE095
C	INTEE096
C TEST FOR NEGATIVE CONCENTRATIONS	INTEE097
DO 9 I=1,LS	INTEE098
IF (C(I) .GE. 0.) GO TO 9	INTEE099
WRITE (6,100)	INTEE100

	NEXT = .TRUE.	INTEE101
	RETURN	INTEE102
	9 CONTINUE	INTEE103
C		INTEE104
C	OPTIONAL AUTOMATIC ELIMINATIONS	INTEE105
	IF (FLIM) CALL AUTO	INTEE106
C		INTEE107
C	GET STEP SIZE FOR NEXT INTEGRATION STEP	INTEE108
	KOUNT = 0	INTEE109
	IF (FRRN .GE. FMAX2) GO TO 10	INTEE110
	IF (HN .GE. HMAX) RETURN	INTEE111
	CALL SEARCH (PERR,EMAX,HN,HNP1,HMAX)	INTEE112
	RETURN	INTEE113
10	IF (FRRN .LE. FMAX56) RETURN	INTEE114
	IF (HN .LE. HMIN) RETURN	INTEE115
	CALL SEARCH (PERR,EMAX,HMIN,HNP1,HN)	INTEE116
	RETURN	INTEE117
C		INTEE118
	END	INTEE119

CINTG	INTEGRAL OF Y*DX--TRAPEZOIDAL/UNEQUAL X S	INTG00001
	SUBROUTINE INTG(Y,X,YDXI,IL,IH)	INTG00002
	DIMENSION Y(1),X(1),YDXI(1)	INTG00003
C*		INTG00004
C*	EVALUATE INTEGRAL--- Y*DX ***TRAPEZOIDAL RULE	INTG00005
C*	*****UNEQUAL SPACING	INTG00006
C*		INTG00007
1	J=IL+1	INTG00008
2	I=I+1	INTG00009
	DX=X(I)-X(I-1)	INTG00010
	TERM=.5*(Y(I)+Y(I-1))	INTG00011
3	YDXI(I)=YDXI(I-1)+TERM*DX	INTG00012
	IF(I.LT. IH) GO TO 2	INTG00013
10	RETURN	INTG00014
	END	INTG00015

CJETINP	JET--INPUT ROUTINE	JETINP01
	SUBROUTINE JETINP	JETINP02
C*****		JETINP03
C*****	SPECIAL VERSION FOR 12-SPECIES	JETINP04
C*****		JETINP05
	LOGICAL TAPIN, TAPOT	JETINP06
	INTEGER PLOT	JETINP07
	LOGICAL MCHANG	JETINP08
	LOGICAL AMBIO	JETINP09
	LOGICAL DPRIN	JETINP10
	LOGICAL EOF, ERR	JETINP11
	LOGICAL AXI, CMPRS, QJET, TURBJ, CORE	JETINP12
	INTEGER XPRN	JETINP13
	LOGICAL ENTRY1	JETINP14
	REAL MJET, ME, MUREF	JETINP15
	REAL MACH	JETINP16
	COMMON /FILK/CSC	JETINP17
	COMMON /SING/ SSD(43)	JETINP18
	COMMON /CARRY/ NEW	JETINP19
	LOGICAL NEW	JETINP20
	COMMON /RSTART/ NREG, RESTRT, NRES, MIXPRE	JETINP21
	LOGICAL MIXPRE	JETINP22
	COMMON /EDGE/ YJETE, SFEDGE	JETINP23
	COMMON /JETINJ/ JINJ, ZDIAD(6), NJD	JETINP24
	COMMON /BCO/ UO, EO, THO	JETINP25
	COMMON /CTRL2/	JETINP26
	* EDGE1, SFI, MERGE, XMERGE, YMERGE,	JETINP27
	* SLOPE1, SLOPE2, CEPT1, CEPT2	JETINP28
	COMMON /MERGET/ MER, MERSTP, XMRG	JETINP29
	LOGICAL TWO, MERGE, MER, MERSTP	JETINP30
	COMMON /PROPJ2/ MACHD, REFLO, Y1, YD, MERGP	JETINP31
	REAL MACHD, MJETO	JETINP32
	COMMON /MISC/ PM(10), PLOT	JETINP33
	COMMON /INPJ/ ENTRY1	JETINP34
	COMMON /UMESH/MCG(7)	JETINP35
C*		JETINP36
C*****	INPUT COMMON	JETINP37
C*		JETINP38
	COMMON /INPJET/	JETINP39
	* ZDIAD(11), JAXI, NJ, NY, X(100), XPRN(100), GAM, ZRG(6)	JETINP40
C*		JETINP41
C*****	CONTROL COMMON	JETINP42
C*		JETINP43
	COMMON /CTRL/	JETINP44
	* NXTA, CMPRS, QJET, TURBJ, COEF(10)	JETINP45
	* NPV, NP2, DXC, XV, XDD	JETINP46
	* DSTOR(800)	JETINP47
C*		JETINP48
C*****	PROFILE COMMON	JETINP49
C*		JETINP50

C*	COMMON /PROF/ PSI(200),Y(200),UJ(200),THD(200),ED(200)	JETINP51
C*****	CONSTANT AND ERROR COMMON	JETINP52
C*		JETINP53
C*	COMMON /CNERR/ BITS , ERR , GC , GCJ , FOOT	JETINP54
C*		JETINP55
C*****	BOUNDARY CONDITION COMMON	JETINP56
C*		JETINP57
C*	COMMON /BC/ UEDGE , EFEDGE , THEDGE	JETINP58
C*		JETINP59
C*****	POTENTIAL CORE COMMON	JETINP60
C*		JETINP61
C*	COMMON /CURED/ XCORE , CORE , CORSTP	JETINP62
C*		JETINP63
C*****	SCALER (UNITS CONVERSION) COMMON	JETINP64
C*		JETINP65
C*	COMMON /SCALER/ SP , SV , SLEN	JETINP66
C*		JETINP67
C*****	JET PROPERTIES COMMON	JETINP68
C*		JETINP69
C*	COMMON /JET/	JETINP70
*	BB(100),UC(100),TC(100),TIC(100),	JETINP71
*	PTC(100) , WJ(100) , YJ(100) ,	JETINP72
*	YSJNIC(100)	JETINP73
COMMON	/JET1/ FLOWJ,TTO,NX,EJET	JETINP74
COMMON	/PROPT/	JETINP75
*	P , PRL , PRT , RGAS , SCC ,	JETINP76
*	TREF , VSREF , MACH , XLC ,	JETINP77
*	XREFL,CDIFF,CHI,DUM(1401)	JETINP78
COMMON	/XPRIN/ DPRIN	JETINP79
COMMON	/CPRJP/ CT1,CT2,CT3,CT4,CT5,CT6,CT7,CT8,CT9,CT10	JETINP80
COMMON	/CPRJP2/ CTP, CTS , CTM	JETINP81
COMMON	/RATIO/ AMBTO	JETINP82
COMMON	/IDFILE/ TAPIN,TAPOT	JETINP83
COMMON	/PARAM/ B(500),DIAJ(1),MJET,TJET,PTJET,VJET,	JETINP84
*	TJET,PE,VE,ME,TIE,TE,AXI,RG(1),PR,PRT,SC,TREF,	JETINP85
*	MUREF,TW,DIAO(1),MJETO,TJETO,VJETO,PTJETO,	JETINP86
*	TJETO,MCHANG(1),CK,DYI,NMSH,CXPC,CXTP,NRED,	JETINP87
*	MIX,SUPR,MAXIT,TOL,CF,NB(1),TAB(5),ND(1),TAD(4),	JETINP88
*	D(800),DUMINP(461)	JETINP89
C*		JETINP90
COMMON	/MIXER/ JMIX,RD(100),XD(100),ZCF,YR(100)	JETINP91
LOGICAL	MIX	JETINP92
COMMON	/FLORAL/ MMAXIT,JSUP,NIT,PSID,YDD,YDC,	JETINP93
*	P1,P2,UCL,ZTOL,UPSTRM,CVG	JETINP94
LOGICAL	SUPR,CVG,UPSTRM	JETINP95
COMMON	/ACONVG/ YCD(100),PD(100),INDC(100), CHOKE, CHOKED	JETINP96
LOGICAL	CHKE, CHOKED	JETINP97
COMMON	/OFIT/ CLSP(100)	JETINP98
COMMON	/STA2/ MACH2,TS2,SS2,V2,RM02,DPOX2	JETINP99
		JETINP00

REAL MACH2	JETINP01
COMMON /BCMIX2/ GRADU,TW,MUW,RHOW,PTE,TIE	JETINP02
REAL MUW	JETINP03
COMMON /CBNDY/ YCB(100),CLSPCB(100),YCB1 , UCL1	JETINP04
COMMON /OUTMIX/ NXORIG	JETINP05
COMMON /INNAME/ MNB(6),MND(5)	JETINP06
COMMON /TAG/ DUMID(40) , IDENT(10)	JETINP07
COMMON /DIFEQI/	JETINP08
* NC , CNAME(12) , ALJ(12) , ALJO(12) , ALE(12) , SCW(12) ,	JETINP09
* YCPRF(12) , HCPRF(12) , CPC(3,12)	JETINP10
COMMON /DICTRL/ DIFF , CND(10)	JETINP11
LOGICAL DIFF	JETINP12
COMMON /MOLES / ALX(100,12)	JETINP13
COMMON /BCMOL / ALEDGE(12),ALO(12)	JETINP14
COMMON /JET3/ STADD,NV,STATF	JETINP15
COMMON /LLTERP/ LTERP	JETINP16
LOGICAL LTERP	JETINP17
COMMON /SCALED/ SCLO,ALXLIM	JETINP18
LOGICAL SCLO	JETINP19
COMMON /CRMDD / CJRMDD	JETINP20
C*	JETINP21
DIMENSION IMP(10)	JETINP22
EQUIVALENCE (IMP(1),PM(1))	JETINP23
DIMENSION XPRN1(100)	JETINP24
EQUIVALENCE (XPRN1(1),XPRN(1))	JETINP25
DIMENSION ICHANG(1)	JETINP26
DIMENSION DMOL(6)	JETINP27
EQUIVALENCE (ICHANG(1),MCHANG(1))	JETINP28
EQUIVALENCE (IAXI,AXI),(ITWO,TWO),(ISUP,SUPB),(IMIX,MIX)	JETINP29
EQUIVALENCE (IBITS,BITS)	JETINP30
DIMENSION TID(200)	JETINP31
EQUIVALENCE (TID(1),ED(1))	JETINP32
NAMELIST /A/ DIAJ,MJET,TJET,PTJET,VJET,	JETINP33
* TIJET	JETINP34
* PE , VE , ME , TIE , TE	JETINP35
* AXI , NJ , NM	JETINP36
* GAM , RG , PR , PRT	JETINP37
* X , XPRN	JETINP38
* SC , TREF , MUREF	JETINP39
* SP , SV , SLEN	JETINP40
* DPRIN , PLOT	JETINP41
* NRED , PM	JETINP42
* CXPC , CXTP	JETINP43
* MCHANG , IDENT	JETINP44
* Y , UD , THD , ED , TID	JETINP45
* CT1,CT2,CT3,CT4,CT5,CT6,CT7,CT8,CT9	JETINP46
* CT4,CT8,CTP,B,VR,TAB,D,ND,TAD,RESTR,	JETINP47
* CK,DY1,NM5H,MIX,MAXIT,CF,TOL,SUPB,RO,XD,YCB,	JETINP48
* TWO , DIAJ , MJETO , TJETO	JETINP49
* VJETO,PTJETO,TIJETO,NJO,	JETINP50

* NC,CNAME,ALJ,ALJO,ALE,SCM,TCPRF,HCPRF,CPC,ALX,DIFF	JETINP51
* ,CSC,NDIGIT,LTERP,CDIFF,CHI,SCLO,ALXLIM , CJRMOD	JETINP52
DATA BLANK/6H	JETINP53
C*	JETINP54
C*	JETINP55
C* READ NAMELIST SA	JETINP56
C*	JETINP57
2 CALL FLGERR(5,ERR)	JETINP58
CALL SETM(1,BITS,Y,800)	JETINP59
Y(1)=BITS	JETINP60
CALL SETM(1,BLANK,IDENT,10)	JETINP61
C*	JETINP62
C* INITIALIZE B ARRAY, DUMMY SINGLE CELL INPUT	JETINP63
C*	JETINP64
IF(.NOT. NEW) GO TO 4091	JETINP65
4000 CALL SETM(2,BITS,B,543,ND,805)	JETINP66
GO TO 4096	JETINP67
4091 CALL MOVE(1,SSD,DIAJ,43,1)	JETINP68
NEW=.TRUE.	JETINP69
CALL SETM(5,BITS,ZDIAJ(2),4,ZDIAJ(8),2,ZDIAD(2),4,B,500,D,800)	JETINP70
4096 IF(TAPIN) CALL JTFIL(3,DUMYF)	JETINP71
3 READ (5,A)	JETINP72
CALL MOVE(1,DIAJ,SSD,43,1)	JETINP73
999 IF(ERR) RETURN	JETINP74
C*	JETINP75
C* RESET POSSIBLE BITS VALUES	JETINP76
C*	JETINP77
IF(IAXI.NE.BITS) IAXI=IAXI	JETINP78
IF(IMIX.NE.BITS) IMIX=IMIX	JETINP79
IF(ITWO.NE.BITS) ITWO=ITWO	JETINP80
IF(ISUP.NE.BITS) ISUP=ISUP	JETINP81
IF(MAXIT.NE.BITS) MAXIT=MAXIT	JETINP82
IF(TOL.NE.BITS) TOL=TOL	JETINP83
IF(CF.NE.BITS) ZCF=CF	JETINP84
IF(CXPC.EQ.BITS .AND. TWO) CXPC=.04	JETINP85
IF(CXTP.EQ.BITS .AND. TWO) CXTP=.04	JETINP86
IF(CXPC.EQ.BITS .AND. MIX) CXPC=.05	JETINP87
IF(CXTP.EQ.BITS .AND. MIX) CXTP=.05	JETINP88
4001 DO 4010 L=1,11	JETINP89
IF(L.GT.6) GO TO 4002	JETINP90
IF(RG(L).NE.BITS) ZRG(L)=RG(L)	JETINP91
IF(DIAD(L).NE.BITS) ZDIAD(L)=DIAD(L)	JETINP92
4002 IF(L.GT.7) GO TO 4003	JETINP93
IF(ICHANG(L).NE.BITS) MCG(L)=ICHANG(L)	JETINP94
4003 IF(DIAJ(L).NE.BITS) ZDIAJ(L)=DIAJ(L)	JETINP95
IF(NH(L).NE.BITS) MNH(L)=NH(L)	JETINP96
4010 CONTINUE	JETINP97
C*	JETINP98
C* MOVE DATA BACK TO DUMMY ARRAY	JETINP99
C*	JETINP00

4020 CALL MOVE(5,ZDIAJ,DIAJ,11,1,MCG,MCHANG,7,1,	JETINP01
* ZDIAJ,DIAJ,6,1,ZRG,RG,6,1,MNB,NB,11,1)	JETINP02
IAXI=JAXI	JETINP03
ITWO=JTNQ	JETINP04
ISUP=JSUP	JETINP05
IMIX=JMIX	JETINP06
CF=ZCF	JETINP07
TOL=ZTOL	JETINP08
MAXIT=MMAXIT	JETINP09
IF(MIX,AND,TWO) GO TO 444	JETINP10
IF(TWO) CT1=.175	JETINP11
IMP(1)=IFIX(PH(1))	JETINP12
CMPS=.FALSE.	JETINP13
KGO=2	JETINP14
C*	JETINP15
C* MOVE INPUT ARRAYS TO STORAGE	JETINP16
C*	JETINP17
4030 CONTINUE	JETINP18
C*	JETINP19
C* COUNT NUMBER OF AXIAL STATIONS	JETINP20
C*	JETINP21
C*	JETINP22
C* IF MIXER NOZZLE CASE, NON-DIMENSIONALIZE	JETINP23
C* AND CURVE FIT DUCT CO-ORDINATES	JETINP24
C*	JETINP25
IF(RESTRT.NE,BITS) GO TO 4300	JETINP26
IF(.NOT,MIX) GO TO 4442	JETINP27
DO 4440 L=1,100	JETINP28
IF(XD(L).EQ,HITS) GO TO 4445	JETINP29
4440 CONTINUE	JETINP30
ERR=.TRUE.	JETINP31
GO TO 999	JETINP32
C*	JETINP33
4445 NXTA=L-1	JETINP34
C* FILL UNFILLED ARRAYS	JETINP35
C*	JETINP36
4040 CALL FILL(XD,RD,1,NXTA)	JETINP37
CALL FILL(XD,YCB,1,NXTA)	JETINP38
TERMD=1,/DIAJ	JETINP39
CALL FMPYC(1,TERMD,XD,X,NXTA)	JETINP40
TERMD=2,/DIAJ	JETINP41
CALL FMPYC(2,TERMD,RD,YR,NXTA,YCB,YCB,NXTA)	JETINP42
C*	JETINP43
CALL LCFIT(XD,YR,NXTA,1,XD(1),Y99,1,0,CLSP)	JETINP44
CALL LCFIT(XD,YCB,NXTA,1,XD(1),YCB1,1,0,CLSPCB)	JETINP45
GO TO 6	JETINP46
4442 DO 4 L=1,100	JETINP47
IF(X(L).EQ,BITS) GO TO 5	JETINP48
4 CONTINUE	JETINP49
444 ERR=.TRUE.	JETINP50

GO TO 999	JETINP51
5 NXTA=L-1	JETINP52
GO TO 6	JETINP53
C*	JETINP54
C* SPECIAL X-TABLE PROCESSING ON RESTART	JETINP55
C*	JETINP56
4300 NXX=NXTA+1	JETINP57
NXD=NV+1	JETINP58
XLOC=RESTR	JETINP59
IF(MIX) RESTR=RESTR/DIAJ	JETINP60
C*	JETINP61
C* SEARCH FOR RESTART LOC.	JETINP62
C*	JETINP63
DO 4310 L=1,100	JETINP64
IF(X(L).EQ. RESTR) NRES=L	JETINP65
IF(X(L).EQ. BITS) GO TO 4320	JETINP66
4310 CONTINUE	JETINP67
4311 ERR=.TRUE.	JETINP68
GO TO 999	JETINP69
4320 NXTA=L-1	JETINP70
IF(NRES .EQ. 1) GO TO 4311	JETINP71
C*	JETINP72
IF(.NOT. MIX) .AND. MIXPRE) GO TO 4329	JETINP73
IF(.NOT. MIX) GO TO 6	JETINP74
DO 4340 L=1,100	JETINP75
IF(XD(L).EQ. BITS) GO TO 4350	JETINP76
4340 CONTINUE	JETINP77
4350 NXORIG=L-1	JETINP78
CALL FILL(XD(NXD),RD(NXD),NXD,NXORIG)	JETINP79
CALL FILL(XD(NXD),YCB(NXD),NXD,NXORIG)	JETINP80
TERMD=1./DIAJ	JETINP81
NUM=NXORIG+1-NXD	JETINP82
CALL FMPYC(1,TERMD,XD(NXD),X(NXX),NUM)	JETINP83
TERMD=2.*TERMD	JETINP84
CALL FMPYC(2,TERMD,RD(NXD),YR(NXD),NUM,YCB(NXD),YCB(NXD),NUM)	JETINP85
CALL LCFIT(XD,YR,NXORIG,1,XD(1),Y99,1,0,CLSP)	JETINP86
CALL LCFIT(XD,YCB,NXORIG,1,XD(1),YCB1,1,0,CLSPBC)	JETINP87
NXTA=NXX+NUM-1	JETINP88
4329 DO 4328 L=1,NXTA	JETINP89
IF(XLOC .EQ. XD(L)) PF=PD(L)	JETINP90
4328 CONTINUE	JETINP91
C*	JETINP92
C* TEST FOR COMPRESSIBILITY,TRANSPORT OF Q--INITIALIZE	JETINP93
C*	JETINP94
6 IF(MJET.NE.BITS) CMPRS=.TRUE.	JETINP95
IF(PIJET.NE. BITS) CMPRS=.TRUE.	JETINP96
IF(.NOT. CMPRS) CALL SETM(1,BITS,PTC,100)	JETINP97
IF(MIX .AND. (RESTR.EQ. BITS)) NXORIG=NXTA	JETINP98
C*	JETINP99
C* SET PRINT INDICATOR	JETINP00

C*		JETINP01
4050	IF((XPRN(1),EQ,2) .OR. (XPRN(1),EQ,2.)) CALL SETM(1,1,XPRN(2),	JETINP02
*	NXTA=1)	JETINP03
	IF((XPRN(1) .EQ. (-2)) .OR. (XPRN(1) .EQ. (-2.)))	JETINP04
X	CALL SETM(1,-1,XPRN(2),NXTA=1)	JETINP05
	DO 4060 L=2,NXTA.	JETINP06
	IF(XPRN(L),EQ,IBITS) XPRN(L)=0	JETINP07
	IF(XPRN(L) .EQ. 0.) GO TO 4060	JETINP08
	IF(IABS(XPRN(L)) .EQ. 1) GO TO 4060	JETINP09
	IF(XPRN(L) .LT. 0.) GO TO 4055	JETINP10
	XPRN(L)=1	JETINP11
	GO TO 4060	JETINP12
4055	XPRN(L)=-1	JETINP13
4060	CONTINUE	JETINP14
	IF(TIJET,NE,0.) GO TO 10	JETINP15
	TURBJ=.FALSE.	JETINP16
C*		JETINP17
C*		JETINP18
C*		JETINP19
C*		JETINP20
	10 IF(CMPRS) GO TO 12	JETINP21
C*		JETINP22
C*	VJET,VE,TJET,TE ASSUMED GIVEN	JETINP23
C*		JETINP24
	MJET=0.	JETINP25
	ME=0.	JETINP26
	CP=GAMH(TJET)	JETINP27
	CPJ = CP	JETINP28
	EJET=1.5/GCJ*TIJET**2*VJET**2	JETINP29
	IF(TWD) EJETO=1.5/GCJ*TIJETO**2*VJET**2	JETINP30
	TTE=TE	JETINP31
	PTF=PE*(1.+S*VE**2/(GC*RG*TE))	JETINP32
	GO TO 150	JETINP33
12	GAM=GAMH(TE)	JETINP34
	VSE=SQRT(GAM*RG*GC*TE)	JETINP35
	IF(ME,EQ,BITS) GO TO 122	JETINP36
	VE=ME*VSE	JETINP37
	GO TO 125	JETINP38
122	ME=VE/VSE	JETINP39
125	TRM=1.+S*(GAM-1.)*ME**2	JETINP40
	TTE=TE*TRM	JETINP41
	PTE=PE*(TRM)**(GAM/(GAM-1.))	JETINP42
	IF(MJET,EQ,BITS) KGO=1	JETINP43
	IF(PTJET,EQ,BITS) KGO=2	JETINP44
	GO TO (130,140),KGO	JETINP45
C*		JETINP46
	140 IF(TJET,NE,BITS) GO TO 141	JETINP47
C*		JETINP48
C*	DETERMINE TJET (GAM=GAM(T))	JETINP49
C*		JETINP50

TLT=VJET*VJET/(RG*GC*MJET*MJET)	JETINP51
TJET=(TLT/2,23708)**(1./,92979)	JETINP52
IF(TJET,GT. 800. .AND. TJET,LT.3600.) GO TO 1410	JETINP53
GAM=1.4	JETINP54
IF(TJET,GE.3600.) GAM=1.254	JETINP55
GO TO 1411	JETINP56
1410 GAM=GAMH(TJET)	JETINP57
1411 GAM1=GAM/(GAM-1.)	JETINP58
GAM2=.5*(GAM-1.)	JETINP59
GAM3=1./GAM1	JETINP60
CP=GAM1*RG*GC/GCJ	JETINP61
GO TO 142	JETINP62
141 GAM=GAMH(TJET)	JETINP63
VJET=MJET*SQRT(GAM*RG*GC*TJET)	JETINP64
GO TO 1411	JETINP65
142 EJFT=1.5/GCJ*TIJET*TIJET*VJET*VJET	JETINP66
PTJET=PE*(1.+GAM2*MJET**2)**GAM1	JETINP67
GO TO 150	JETINP68
C*	JETINP69
130 IF(VJET,EQ.8ITS) GO TO 131	JETINP70
ASSIGN 1312 TO KPGO	JETINP71
GAMG = 1.38	JETINP72
IT = 0	JETINP73
PRAT = PTJET/PE	JETINP74
1301 IT = IT+1	JETINP75
GAMG1 = (GAMG-1.)/GAMG	JETINP76
TJET = VJET**2*(GAMG-1.)/(2.*GAMG*RG*GC*(PRAT**GAMG1-1.))	JETINP77
IF(TJET,GE.800. .AND. TJET,LT.3600.) GO TO 1302	JETINP78
GAM = 1.4	JETINP79
IF(TJET,GE.3600.) GAM=1.254	JETINP80
GO TO 1303	JETINP81
1302 GAM = GAMH(TJET)	JETINP82
1303 IF((ABS(GAM-GAMG)),LE. .001 .OR. IT,GE.10) GO TO 1311	JETINP83
GAMG = GAM	JETINP84
GO TO 1301	JETINP85
1310 GAM=GAMH(TJET)	JETINP86
1311 GAM1=GAM/(GAM-1.)	JETINP87
GAM2=.5*(GAM-1.)	JETINP88
GAM3=1./GAM1	JETINP89
CP=GAM1*RG*GC/GCJ	JETINP90
TTRJ=(PTJET/PE)**GAM3	JETINP91
GO TO KPGO, (1312,1364)	JETINP92
1312 EJFT=1.5/GCJ*TIJET*TIJET*VJET*VJET	JETINP93
MJET=VJET/SQRT(GAM*RG*GC*TJET)	JETINP94
GO TO 150	JETINP95
131 ASSIGN 1364 TO KPGO	JETINP96
GO TO 1310	JETINP97
1364 MJET=SQRT((TTRJ-1.)/GAM2)	JETINP98
VJET=MJET*SQRT(GAM*RG*GC*TJET)	JETINP99
EJET=1.5/GCJ*TIJET*TIJET*VJET*VJET	JETINP00

C*		JETINP01
150	CPJ=CP	JETINP02
C*		JETINP03
	IF(,NOT, TWO) GO TO 151	JETINP04
C**	LOGIC FOR FLOW CONDITIONS OF OUTER JET	JETINP05
C**		JETINP06
	IF(CMPRS) GO TO 160	JETINP07
	MJETO=0,	JETINP08
	GO TO 151	JETINP09
160	IF(MJETO.EQ,BITS) KMO=1	JETINP10
	IF(PTJETO.EQ,BITS) KMO=2	JETINP11
	GO TO (170,165) , KMO	JETINP12
165	IF(TJETO.NE,BITS) GO TO 168	JETINP13
C*		JETINP14
C*	DETERMINE TJETO (GAMO=GAMO(T))	JETINP15
C*		JETINP16
	TLT=VJETO*VJETO/(RG*GC*MJETO*MJETO)	JETINP17
	TJF10=(TLT/2.23708)**(1./,92979)	JETINP18
	IF(TJETO.GT,800. .AND. TJETO.LT,3600.) GO TO 1610	JETINP19
	GAMO=1.4	JETINP20
	IF(TJETO.GE,3600.) GAM=1.254	JETINP21
	GO TO 1611	JETINP22
1610	GAMO=GAMH(TJETO)	JETINP23
1611	GAM1=GAM/(GAM-1.)	JETINP24
	GAM2=.5*(GAM-1.)	JETINP25
	GAM3=1./GAM1	JETINP26
	CP=GAM1*RG*GC/GCJ	JETINP27
	GO TO 169	JETINP28
168	GAMO=GAMH(TJETO)	JETINP29
	VJETO=MJETO*SQRT(GAMO*RG*GC*TJETO)	JETINP30
	GO TO 1611	JETINP31
169	EJETO=1.5/GCJ*TJETO*TJETO*VJET*VJET	JETINP32
	PTJETO=PE*(1.+GAM2*MJETO**2)**GAM1	JETINP33
	GO TO 151	JETINP34
C*		JETINP35
C*		JETINP36
170	IF(VJETO.EQ, BITS) GO TO 171	JETINP37
	ASSIGN 1712 TO KPGD	JETINP38
	GAMG = 1.38	JETINP39
	IT = 0	JETINP40
	PRAT = PTJETO/PE	JETINP41
1701	IT = IT+1	JETINP42
	GAMG1 = (GAMG-1.)/GAMG	JETINP43
	TJETO = VJETO**2*(GAMG-1.)/(2.*GAMG*RG*GC*(PRAT**GAMG1-1.))	JETINP44
	IF(TJETO.GE,800. .AND. TJETO.LT,3600.) GO TO 1702	JETINP45
	GAMO = 1.4	JETINP46
	IF(TJETO.GE,3600.) GAMO=1.254	JETINP47
	GO TO 1703	JETINP48
1702	GAMO = GAMH(TJETO)	JETINP49
1703	IF((ABS(GAMG-GAMO)).LE. .001 .OR. IT.GE.10) GO TO 1711	JETINP50

GAMG = GAMO	JETINP51
GO TO 1701	JETINP52
1710 GAMJ=GAMH(TJETO)	JETINP53
1711 GAM1=GAMJ/(GAMO-1.)	JETINP54
GAM2=.5*(GAMO-1.)	JETINP55
GAM3=1./GAM1	JETINP56
CP=GAM1*RG*GC/GCJ	JETINP57
TIRJN=(PTJETO/PE)**GAM3	JETINP58
GO TO KPGO,(1712,1764)	JETINP59
1712 EJETD=1.5/GCJ*TIJFT ₇ *TIJETO*VJET*VJET	JETINP60
MJETD=VJETD/SQRT(GAMO*RG*GC*TJETO)	JETINP61
GO TO 151	JETINP62
171 ASSIGN 1764 TO KPGO	JETINP63
GO TO 1710	JETINP64
1764 EJETD=1.5/GCJ*TIJETO*TIJETO*VJET*VJET	JETINP65
MJETD=SQRT((TIRJN-1.)/GAM2)	JETINP66
VJETD=MJETD*SQRT(GAM*RG*GC*TJETO)	JETINP67
151 IF(TWO) GO TO 152	JETINP68
IF(TJET.EQ.TE) QJET=.FALSE.	JETINP69
GO TO 153	JETINP70
152 IF(TJET.EQ.TE .AND. TJET.EQ.TJETO) QJET=.FALSE.	JETINP71
153 CONTINUE	JETINP72
IF(.NOT. QJET) CALL SETM(1,1.,TC,100)	JETINP73
EE=1.5/GCJ*TE*TE*VJET*VJET	JETINP74
TTO=TJET+.5*VJET*VJET/(GCJ*CPJ)+EJET/CPJ	JETINP75
AMHTO=.FALSE.	JETINP76
DIBO=ARS(TTO-TE)	JETINP77
IF(DIBO.LT..5) AMHTO=.TRUE.	JETINP78
CALL MOVE(3,DIAJ,ZDIAJ,6,1,DIAJ,ZDIAJ,11,1,MCHANG,MCG,7,1)	JETINP79
C* INITIALIZE PROFILES	JETINP80
C* 20 NJP=NJ+1	JETINP81
NJM=NJ-1	JETINP82
C* BOUNDARY CONDITIONS--V/VJET=VE/VJET,THETA=0,	JETINP83
C* E/EJET=EE/EJET	JETINP84
C* 23 THEDGE=TE/TJET	JETINP85
IE(TURBJ) EEDGE=EE/EJET	JETINP86
UEEDGE=VE/VJET	JETINP87
C* MESH DEFINITION AT INITIAL STATION-- IF Y(1).NE.BITS,	JETINP88
C* INPUT PROFILES USED AS GIVEN	JETINP89
C* NJ=CURRENT SPECIFIED MESH NUMBER OF JET CORNER	JETINP90
C* UO=1.	JETINP91
THO=1.	JETINP92
IF(TURBJ) EO=1.	JETINP93
	JETINP94
	JETINP95
	JETINP96
	JETINP97
	JETINP98
	JETINP99
	JETINP00

	DO 2310 LL=1,NC	JETINP01
	ALEDGE(LL)=ALE(LL)	JETINP02
2310	ALO(LL)=ALJ(LL)	JETINP03
	IF(,NOT, TWO) GO TO 2313	JETINP04
	THO=TJETD/TJET	JETINP05
	UO=VJETD/VJET	JETINP06
	IF(TURBJ) EO=EJETD/EJET	JETINP07
	DO 2311 LL=1,NC	JETINP08
2311	ALO(LL)=ALJO(LL)	JETINP09
2313	IGO=1	JETINP10
	IF(RESTRT,NE,BITS) GO TO 3840	JETINP11
	IF(Y(1),NE,BITS) IGO=2	JETINP12
	GO TO (21,30), IGO	JETINP13
C*		JETINP14
C*	GENERATE INITIAL PROFILES	JETINP15
C*		JETINP16
21	LGO=1	JETINP17
	LOW=2	JETINP18
	NE=NJ+3	JETINP19
	Y(1)=0.	JETINP20
	IF(MIX) Y(1)=YCB1	JETINP21
	UD(1)=1.	JETINP22
	THD(1)=1.	JETINP23
	IF(TURBJ) ED(1)=1.	JETINP24
C*		JETINP25
	DYI=(1.-Y(1))/FLOAT(NJ-1)	JETINP26
	DY2=.5*DYI	JETINP27
	LHO=1	JETINP28
	IF(TWO) LHO=2	JETINP29
	IF(,NOT,DIFF) GO TO 2161	JETINP30
	DO 2160 LL=1,NC	JETINP31
2160	ALX(1,LL) = ALJ(LL)	JETINP32
2161	GO TO (22,222),LHO	JETINP33
22	DU=UEDGE-UO	JETINP34
	DTH=THEDGE-THO	JETINP35
	IF(TURBJ) DE=EEDGE-EO	JETINP36
	IF(,NOT,DIFF) GO TO 26	JETINP37
	DO 2170 LL=1,NC	JETINP38
2170	D4OL(LL)=ALEDGE(LL)-ALO(LL)	JETINP39
	GO TO 26	JETINP40
2222	DU=UO-1.	JETINP41
	DTH=THO-1.	JETINP42
	IF(TURBJ) DE=EO-1.	JETINP43
	IF(,NOT,DIFF) GO TO 26	JETINP44
	DO 2171 LL=1,NC	JETINP45
2171	D4OL(LL)=ALO(LL)-ALJ(LL)	JETINP46
26	DO 27 L=LOW,NE	JETINP47
	IF(L.GT,NJM) GO TO 25	JETINP48
	Y(L)=Y(L-1)+DYI	JETINP49
	UD(L)=UD(LOW-1)	JETINP50

	THD(L)=THD(LDW=1)	JETINP51
	IF(,NOT,TURBJ) GO TO 2619	JETINP52
	ED(L)=ED(LDW=1)	JETINP53
2619	IF(,NOT,DIFF) GO TO 27	JETINP54
	DO 2620 LL=1,NC	JETINP55
2620	ALX(L,LL)=ALX(LDW=1,LL)	JETINP56
	GO TO 27	JETINP57
25	Y(L)=Y(L-1)+DY2	JETINP58
	UD(L)=UD(L-1)+.25*DU	JETINP59
	THD(L)=THD(L-1)+.25*DTM	JETINP60
	IF(,NOT,TURBJ) GO TO 2519	JETINP61
	ED(L)=ED(L-1)+.25*DE	JETINP62
2519	IF(,NOT,DIFF) GO TO 27	JETINP63
	DO 2520 LL=1,NC	JETINP64
2520	ALX(L,LL)=ALX(L-1,LL)+.25*DMOL(LL)	JETINP65
27	CONTINUE	JETINP66
	GO TO (32,2223) , LHO	JETINP67
2223	DIST=DIAD/DIAJ-(1,+DYI)	JETINP68
	DYI=DIST/FLDGT(NJO-NE)	JETINP69
	EDGEI=Y(NE)	JETINP70
	LDW=NE+1	JETINP71
	NE=NJO+3	JETINP72
	NJM=NJO-1	JETINP73
	NJP=NJO+1	JETINP74
	LHO=1	JETINP75
	GO TO 22	JETINP76
32	NRMV=NM=NE	JETINP77
31	NE1=NE+1	JETINP78
	CALL SETM(1,UEdge,UD(NE1),NRMN)	JETINP79
	CALL SETM(1,THEdge,THD(NE1),NRMN)	JETINP80
	IF(TURBJ) CALL SETM(1,EEdge,ED(NE1),NRMN)	JETINP81
	BB(1)=Y(NE)-Y(NJM)	JETINP82
	IF(,NOT,DIFF) GO TO 3119	JETINP83
	DO 3120 LL=1,NC	JETINP84
	ALE1=ALEdge(LL)	JETINP85
	CALL SETM(1,ALE1,ALX(NE1,LL),NRMN)	JETINP86
3120	CONTINUE	JETINP87
3119	GO TO (29,40),LGO	JETINP88
29	DO 28 L=NE1,NM	JETINP89
28	Y(L)=Y(L-1)+DYI	JETINP90
	NPU=NE	JETINP91
	NPD=NE	JETINP92
	UD(NPU)=UEdge	JETINP93
	THD(NPU)=THEdge	JETINP94
	IF(TURBJ) ED(NPU)=EEdge	JETINP95
	IF(,NOT,DIFF) GO TO 35	JETINP96
	DO 2802 LL=1,NC	JETINP97
2802	ALX(NPU,LL)=ALEdge(LL)	JETINP98
	GO TO 35	JETINP99
C*		JETINP00

C*	PROFILES INPUT BY USER	JETINP01
C*		JETINP02
30	DO 36 L=1,NH	JETINP03
	IF(Y(L),EQ,BITS) GO TO 37	JETINP04
36	CONTINUE	JETINP05
	ERR = .TRUE.	JETINP06
	GO TO 999	JETINP07
37	L = L+1	JETINP08
	NE = L	JETINP09
	LGO = 2	JETINP10
	CALL FIL1(Y,UD,1,NE)	JETINP11
	CALL FILL(Y,TID,1,NE)	JETINP12
	CALL FILL(Y,THD,1,NE)	JETINP13
C*	PROFILES DIMENSIONAL-- IN.,FPS.,DEG. R.	JETINP14
	OD = 2./DIAJ	JETINP15
	QV = 1./VJET	JETINP16
	QT = 1./TJET	JETINP17
	CALL FMPYC(1,OD,Y,Y,NE)	JETINP18
	CALL FMPYC(1,QV,UD,UD,NE)	JETINP19
	CALL FMPYC(1,QT,THD,THD,NE)	JETINP20
C		JETINP21
C	CALCULATE IKE FROM INPUT TURBULENT INTENSITY	JETINP22
C		JETINP23
	TKET = 1.5*VJET*VJET/(GCJ+EJET)	JETINP24
	DO 38 L=1,NE	JETINP25
38	ED(L) = TKET*TID(L)**2	JETINP26
	GO TO 32	JETINP27
C*		JETINP28
C*		JETINP29
C*	WIDTH OF MIXING ZONE-- Y(NJP)-Y(NJM))--- 3-MESH POINTS	JETINP30
C*		JETINP31
40	YJETE = Y(NJ)	JETINP32
	UCCL1 = UD(1)	JETINP33
	IF(SCID) GO TO 4111	JETINP34
	DO 41 L=2,NJ	JETINP35
	IF(UD(L),NE,UCCL1) GO TO 42	JETINP36
41	CONTINUE	JETINP37
	ERR = .TRUE.	JETINP38
	GO TO 999	JETINP39
4111	DO 4112 L=1,NJ	JETINP40
	IF(ALX(L,NC),GE,ALXLIM) GO TO 42	JETINP41
4112	CONTINUE	JETINP42
	ERR = .TRUE.	JETINP43
	GO TO 999	JETINP44
42	BB(1) = YJETE-Y(L-1)	JETINP45
	NPU=NE	JETINP46
	NPD=NE	JETINP47
	GO TO 35	JETINP48
3840	CALL JFILE (4,RESTRT)	JETINP49
	NPU=NPD	JETINP50

C*		JETINP51
C*		JETINP52
C*	COMPUTE CONSTANT TERMS IN COEFFICIENTS OF PDEQS	JETINP53
C*		JETINP54
	35 DIA=DIAJ/12,	JETINP55
	YJETF=Y(NPU)	JETINP56
	UCL1=UD(1)	JETINP57
	IF(RESTRT,NE,BITS) YJETE=YJ(NRES)	JETINP58
C*		JETINP59
	COEF(1)=1./(VJET*DIA)	JETINP60
	IF(TURBJ) GO TO 770	JETINP61
	COEF(2)=0,	JETINP62
	GO TO 769	JETINP63
770	COEF(2)=VJET/(GCJ*DIA*EJET)	JETINP64
769	IF(QJET) GO TO 771	JETINP65
	COEF(3)=0,	JETINP66
	COEF(4)=0,	JETINP67
	GO TO 772	JETINP68
771	COEF(3)=EJET/(VJET*DIA*TJET)	JETINP69
	COEF(4)=VJET/(GCJ*DIA*TJET)	JETINP70
772	COEF(5)=DIA/VJET	JETINP71
	COEF(6)=.5*DIA	JETINP72
	COEF(7)=144.*GC/(VJET*VJET)	JETINP73
	COEF(9)=144.*GC/(GCJ*TJET)	JETINP74
C*		JETINP75
C*	INITIALIZE COMMON /PROPJ/	JETINP76
C*		JETINP77
	RGAS=RG	JETINP78
	IF(.NOT.DIFF) GO TO 7772	JETINP79
	MMWT=1545./RGAS	JETINP80
	CND(1)=1./ (MMWT*VJET*DIA*TJET)	JETINP81
7772	MACH=MJET	JETINP82
	P=PE	JETINP83
	VSREF=MUREF	JETINP84
	TRFF=TREF	JETINP85
	SCC=SC	JETINP86
	PRL=PR	JETINP87
	PRTT=PRT	JETINP88
	MACHO=MJETO	JETINP89
	ENTRY1=.TRUE.	JETINP90
	IF(RESTRT,NE,BITS) GO TO 996	JETINP91
	CALL JETPRP	JETINP92
C*		JETINP93
C*		JETINP94
C*	PRINT INITIAL STATION DATA	JETINP95
C*		JETINP96
	996 CALL JTOUT1	JETINP97
	IF(TAPOT,AND,RESTRT,EQ,BITS) CALL JTFILE(1,X(NRES))	JETINP98
C*		JETINP99
	1000 RETURN	JETINP00

END

JETINP01

*JETPRF	GENERATES INITIAL PROFILES FOR -JETMIX-	JETPRF01
SUBROUTINE JETPRF		JETPRF02
COMMON /DPCTRL/ TITLE(20),PRINT(30)		JETPRF03
COMMON /TROUBL/ ERR,ERRMAJ,INERR,PRERR		JETPRF04
LOGICAL ERR,ERRMAJ,INERR,PRERR		JETPRF05
COMMON /JETDAT/ NPTS,RAD(12),IS(12),U(12),SPV(12),MWT(12),CP(12),		JETPRF06
1 FUEL(12),SPALDG(12),TKE(12),OTHER(36)		JETPRF07
COMMON /INDATA/ N,HF,WAR,T2,BETA,T25,FAR5,FIND2C,P0,RCO(11),		JETPRF08
1 RCO2(11),RHC(11),RNOX(11),PT(11),PS(11),BLQC(11),QCO(11)		JETPRF09
COMMON /CREFDA/		JETPRF10
* DIAJ,MJET,TIJET,VJET,GJET(200),PE,TF,TIF,VE,GFX,RG,PR,		JETPRF11
1 PRI,SC,TREF,MUREF,DIFF,NC,CNAME(12),ALF(12),SCM(12),CPC(36),NJ,		JETPRF12
* NM,CT1,CT2,CT3,CT4,CT5,CT6,CT7,CT8,Y(200),UD(200),THD(200),		JETPRF13
* TID(200),ALX(1200)		JETPRF14
COMMON /KEYS/ KEYA(11),KEYB(11),KODA(11),KODR(11)		JETPRF15
COMMON /CPCIFF/ CA,CB,CC,CAA,CBA,CCA		JETPRF16
COMMON /GASCMP/ RICH(24),FOUL(48),ENTH(48),CONC(768),HCINCP(24),		JETPRF17
* OTHER2(192)		JETPRF18
COMMON /CHITS/ BITS,BLANK		JETPRF19
DIMENSION LENI(27),NSA(26)		JETPRF20
DIMENSION RIURE(12),CNAM1(12)		JETPRF21
DIMENSION FD(200)		JETPRF22
EQUIVALENCE (ED(1),TID(1))		JETPRF23
REAL MOLF1(100),MOLF2(100),MOLF3(100),MOLF4(100),		JETPRF24
* MOLF5(100),MOLF6(100),MOLF7(100),MOLF8(100),		JETPRF25
* MOLF9(100),MOLF10(100),MOLF11(100),MOLF12(100)		JETPRF26
EQUIVALENCE (MOLF1,ALX(1)),(MOLF2,ALX(101)),(MOLF3,ALX(201)),		JETPRF27
* (MOLF4,ALX(301)),(MOLF5,ALX(401)),(MOLF6,ALX(501)),		JETPRF28
* (MOLF7,ALX(601)),(MOLF8,ALX(701)),(MOLF9,ALX(801)),		JETPRF29
* (MOLF10,ALX(901)),(MOLF11,ALX(1001)),(MOLF12,ALX(1101))		JETPRF30
REAL N,MASS,MACH,MJET,MUREF		JETPRF31
LOGICAL DIFF		JETPRF32
DIMENSION IJ(12)		JETPRF33
EQUIVALENCE (UJ,U)		JETPRF34
NAMLIST /A/ TIJET,TIE,RG,PR,PRI,SC,TREF,MURFF,NM,		JETPRF35
* CA,CB,CC,CAA,CBA,CCA,CT1,CT2,CT3,CT4,CT5,CT6,CT7,CT8		JETPRF36
C*		JETPRF37
DATA LENI/3,12,1,36,6,100,12,100,9,12,1,36,1,20,1,30,		JETPRF38
* 8,11,2,24,2,48,1,768,1,192,0/		JETPRF39
DATA NSA/3,1HI,1,1HI,6,1HK,1,1HL,9,1HY,1,1HY,		JETPRF40
* 1,2HI1,1,2HI1,8,2HK1,2,2HL1,2,2HM1,1,2HN1,1,2HI2/		JETPRF41
DATA CNAM1 /4HGAS1,4HGAS2,4HGAS3,4HGAS4,4HGAS5,4HGAS6,4HGAS7,		JETPRF42
1 4HGAS8,4HGAS9,5HGAS10,5HGAS11,5HGAS12/		JETPRF43
C*		JETPRF44
C*		JETPRF45
CALL FLGERR(5,INERR)		JETPRF46
READ (5,A)		JETPRF47
IF(INERR) RETURN		JETPRF48
CALL SETM(1,BITS,Y,200)		JETPRF49
CALL MOVE (1,CNAM1,CNAME,12,1)		JETPRF50

RJET=12.*RAD(NPTS+1)	JETPRF51
DIAJ=2.*RJET	JETPRF52
PE=PO	JETPRF53
TE=1S(NPTS+1)	JETPRF54
VE=U(NPTS+1)	JETPRF55
GEX=SPALDG(NPTS+1)	JETPRF56
NC=NPTS+1	JETPRF57
NJ=6*NPTS-1	JETPRF58
DIFF=.TRUE.	JETPRF59
MJET=0.	JETPRF60
VJET=0.	JETPRF61
FLOW=0.	JETPRF62
RTUBE(1)=0.	JETPRF63
PI=3.1415927	JETPRF64
GC=32.174	JETPRF65
DO 120 I=1,NPTS	JETPRF66
IF (I.EQ.NPTS) GO TO 100	JETPRF67
RTUBE(I+1)=SQRT((RAD(I)**2+RAD(I+1)**2)/2.)	JETPRF68
GO TO 110	JETPRF69
100 RTUBF(I+1)=RAD(NPTS+1)	JETPRF70
110 AREA=PI*(RTUBE(I+1)**2-RTUBE(I)**2)	JETPRF71
MASS=AREA*U(I)/SPV(I)	JETPRF72
FLOW=FLOW+MASS	JETPRF73
GAMMA=1.4	JETPRF74
IF (TS(I).GT.(800.)) GAMMA=2.23708/TS(I)**.070271	JETPRF75
IF (TS(I).GE.(3600.)) GAMMA=1.254	JETPRF76
MACH=U(I)/SQRT(GAMMA*GC*RG*TS(I))	JETPRF77
MJET=MJET+MASS*MACH	JETPRF78
VJET=VJET+MASS*U(I)	JETPRF79
120 CONTINUE	JETPRF80
MJET=MJET/FLOW	JETPRF81
VJET=VJET/FLOW	JETPRF82
DO 130 I=1,NPTS	JETPRF83
CPC(3*I-2)=CA	JETPRF84
CPC(3*I-1)=CB	JETPRF85
CPC(3*I)=CC	JETPRF86
SCM(I)=.70	JETPRF87
ALE(I)=0.0	JETPRF88
130 CONTINUE	JETPRF89
NPP1=NPTS+1	JETPRF90
CPC(3*NPP1-2)=CAA	JETPRF91
CPC(3*NPP1-1)=CBA	JETPRF92
CPC(3*NPP1)=CCA	JETPRF93
SCM(NPP1)=.70	JETPRF94
ALE(NPP1)=1.0	JETPRF95
NSL=NJ+3	JETPRF96
NJM1=NJ+1	JETPRF97
CALL SE14 (1,TIJET,TID(1),NJM1)	JETPRF98
NALX=100*NPP1	JETPRF99
CALL SE14 (1,0.0,ALX(1),NALX)	JETPRF00

DO 150 J=NJ,NSL	JETPRF01
IF (J.EQ.NJ) SCL=.90	JETPRF02
IF (J.EQ.NJ+1) SCL=.50	JETPRF03
IF (J.EQ.NJ+2) SCL=.10	JETPRF04
IF (J.EQ.NSL) SCL=0.0	JETPRF05
TID(J)=SCL*TIJET+(1.-SCL)*TIE	JETPRF06
150 CONTINUE	JETPRF07
DRSAVE=.010*HAD(NPTS+1)	JETPRF08
Y(1)=12.*RTUBE(1)	JETPRF09
UD(1) = U(1)	JETPRF10
TMD(1)= TS(1)	JETPRF11
GJFI(1)= SPALDG(1)	JETPRF12
ALX(1)=1.	JETPRF13
DO 200 I=1,NPP1	JETPRF14
J = 6*(I-1)	JETPRF15
DR=DRSAVE	JETPRF16
DRQ2=DR/2.	JETPRF17
IF (I.EQ.NPP1) GO TO 160	JETPRF18
IF (DR.LT.(RTUBE(I+1)-RTUBE(I))/2.) GO TO 160	JETPRF19
DR=(RTUBE(I+1)-RTUBE(I))/4.	JETPRF20
DRQ2=DR/2.	JETPRF21
160 IF (I.EQ.1) GO TO 170	JETPRF22
Y(J)=12.*RTUBE(I)	JETPRF23
UD(J) = 0.5*(U(I-1)+U(I))	JETPRF24
TMD(J)= .5*(TS(I-1)+TS(I))	JETPRF25
GJFI(J)= .5*(SPALDG(I-1)+SPALDG(I))	JETPRF26
K=100*(I-2)+J	JETPRF27
ALX(K)=.50	JETPRF28
K=100*(I-1)+J	JETPRF29
ALX(K)=.50	JETPRF30
JP1=J+1	JETPRF31
Y(JP1)=12.*(RTUBE(I)+DRQ2)	JETPRF32
UD(JP1) = 0.10*U(I-1)+0.90*U(I)	JETPRF33
TMD(JP1)=.10*TS(I-1)+.90*TS(I)	JETPRF34
GJFI(JP1)=.1*SPALDG(I-1)+.9*SPALDG(I)	JETPRF35
K=100*(I-2)+JP1	JETPRF36
ALX(K)=.10	JETPRF37
K=100*(I-1)+JP1	JETPRF38
ALX(K)=.90	JETPRF39
IF (I.EQ.NPP1) GO TO 190	JETPRF40
170 Y(J+2)= 12.*(RTUBE(I)+DR)	JETPRF41
Y(J+3)=12.*(RTUBE(I)+RTUBE(I+1))/2.	JETPRF42
Y(J+4)=12.*(RTUBE(I+1)-DR)	JETPRF43
JP2=J+2	JETPRF44
JP4=J+4	JETPRF45
DO 180 J=JP2,JP4	JETPRF46
UD(J) = U(I)	JETPRF47
TMD(J)=TS(I)	JETPRF48
GJFI(J)= SPALDG(I)	JETPRF49
K=100*(I-1)+J	JETPRF50

	ALX(K)=1.0	JETPRF51
180	CONTINUE	JETPRF52
	J=JP4+1	JETPRF53
	Y(J)=12,*(RTUBE(I+1)-DRQ2)	JETPRF54
	UD(J) = 0.90*U(I)+0.10*U(I+1)	JETPRF55
	THD(J)=.90*TS(I)+.10*TS(I+1)	JETPRF56
	GJET(J)=.9*SPALDG(I)+.1*SPALDG(I+1)	JETPRF57
	K=100*(I-1)+J	JETPRF58
	ALX(K)=.90	JETPRF59
	K=100*J+J	JETPRF60
	ALX(K)=.10	JETPRF61
	GO TO 200	JETPRF62
190	J=J+2	JETPRF63
	Y(J)=12,*(RTUBE(I)+DR)	JETPRF64
	UD(J) = U(I)	JETPRF65
	THD(J)=TS(I)	JETPRF66
	GJET(J)= SPALDG(I)	JETPRF67
	K=100*(I-1)+J	JETPRF68
	ALX(K)=1.0	JETPRF69
200	CONTINUE	JETPRF70
	LEN1(2)=NPP1	JETPRF71
	LEN1(4)=3*NPP1	JETPRF72
	LEN1(6)=NSL	JETPRF73
	LEN1(8)= NSL	JETPRF74
	LEN1(10)=NPP1	JETPRF75
	LEN1(12)=1	JETPRF76
	WRITE (1)	JETPRF77
	* DIAJ,MJET,IJET,VJET,PE,TE,TIE,VE,GEX,RG,PR,PRT,SC,TREF,	JETPRF78
	* MUREF,DIF,NC,CNAME,ALF,SCM,CPC,NJ,NM,CT1,CT2,CT3,CT4,CT5,	JETPRF79
	* CT6,CT7,CT8,GJET,Y,UD,THD,TID,ED,	JETPRF80
	* MOLF1,MOLF2,MOLF3,MOLF4,MOLF5,MOLF6,	JETPRF81
	* MOLF7,MOLF8,MOLF9,MOLF10,MOLF11,MOLF12,	JETPRF82
	* NPTS,RAD,TS,UJ,SPV,MAT,CP,FUEL,SPALDG,TKE,OTHER,	JETPRF83
	* TITLE,PRINT,N,HF,WAR,T2,BF1A,I25,FARS,EINO2C,P0,	JETPRF84
	* RCU,RCU2,RHC,RNOX,PT,PS,BLOC,QCO,	JETPRF85
	* RICH,HCINCP,FUUL,ENTH,CONC,OTHER2	JETPRF86
	* ,GEX	JETPRF87
	RETURN	JETPRF88
	END	JETPRF89

CJETPRP	PROPERTIES OF JET	JETPRP01
	SUBROUTINE JETPRP	JETPRP02
	LOGICAL SUPERT	JETPRP03
	LOGICAL ENTRY1	JETPRP04
	LOGICAL AXI , XPRN , CMPRS , QJET , TURBJ , CORE	JETPRP05
	LOGICAL SUBSON,IMACH	JETPRP06
	LOGICAL ERR	JETPRP07
	LOGICAL AMBTO	JETPRP08
	REAL MJET , ME , MUREF	JETPRP09
	REAL MACH	JETPRP10
	REAL KCP,MUEFF,MUL	JETPRP11
	COMMON /JETTHQ/	JETPRP12
	* TWO , DIAO , MJETO , TJETO , VJETO ,	JETPRP13
	* PTJETO , TIJETO , NJO	JETPRP14
	REAL MJETO,MACHO	JETPRP15
	COMMON /RCO/ UO, EO, THO	JETPRP16
	COMMON /CTRL2/	JETPRP17
	* EDGE1 , SEI , MERGE , XMERGE , YMERGE ,	JETPRP18
	* SLOPE1 , SLOPED , CEPTI , CEPTO	JETPRP19
	COMMON /MERGET/ MER, MERSTP , XMRG	JETPRP20
	LOGICAL TWO, MERGE , MER , MERSTP	JETPRP21
	COMMON /PROPJ2/ MACHO,REFLO,YI,YO,MERGP	JETPRP22
	LOGICAL MERGP	JETPRP23
	COMMON /RATIO/ AMBTO	JETPRP24
	COMMON /INPI/ ENTRY1	JETPRP25
C*****	INPUT COMMON	JETPRP26
C*		JETPRP27
	COMMON /INPJET/	JETPRP28
	* DIAJ , MJET , TJET , PTJET , VJET ,	JETPRP29
	* TIJET ,	JETPRP30
	* PE , VE , ME , TIF , TE ,	JETPRP31
	* AXI , NJ , NH ,	JETPRP32
	* X(100) , XPRN(100) ,	JETPRP33
	* GAM , RG , PR , PRT ,	JETPRP34
	* SC , TREF , MUREF	JETPRP35
C*		JETPRP36
C*****	CONTROL COMMON	JETPRP37
C*		JETPRP38
	COMMON /CTRL/	JETPRP39
	* NXIA , CMPRS , QJET , TURBJ , COEF(10) ,	JETPRP40
	* NPU , NPD , DXC , XU , XDD ,	JETPRP41
	* DSTOR(800)	JETPRP42
C*		JETPRP43
C*****	PROFILE COMMON	JETPRP44
C*		JETPRP45
	COMMON /PROF/ PSI(200),YO(200),UD(200),THO(200),EO(200)	JETPRP46
C*		JETPRP47
C*****	CONSTANT AND ERROR COMMON	JETPRP48
C*		JETPRP49
	COMMON /CERR/ BITS , ERR , GC , GCJ , FOOT	JETPRP50

C*		JETPRP51
C*****	BOUNDARY CONDITION COMMON	JETPRP52
C*		JETPRP53
	COMMON /BC/ UEDGE , FEDGE , THEDGE	JETPRP54
C*		JETPRP55
C*****	POTENTIAL CORE COMMON	JETPRP56
C*		JETPRP57
	COMMON /CORES/ XCORE , CORE , CORSTP	JETPRP58
C*		JETPRP59
C*****	SCALER (UNITS CONVERSION) COMMON	JETPRP60
C*		JETPRP61
	COMMON /SCALER/ SP , SV , SLEN	JETPRP62
C*		JETPRP63
C*****	JET PROPERTIES COMMON	JETPRP64
C*		JETPRP65
	COMMON /JET/	JETPRP66
*	B(100) , UC(100) , TC(100) , TIC(100) ,	JETPRP67
*	PTC(100) , WJ(100) , YJ(100) ,	JETPRP68
*	YSONIC(100)	JETPRP69
	COMMON /JET1/ FLOWJ,ITO,NX,EJET	JETPRP70
	COMMON /JET2/ TTC(100)	JETPRP71
	COMMON /PROPJT/	JETPRP72
*	P , PRL , PRTT , RGAS , SCC ,	JETPRP73
*	TREFF , VSREF , MACH , XLC ,	JETPRP74
*	RFFL , C , CHI , RNORM ,	JETPRP75
*	RHO(200) , MUL(200) , KCP(200) ,	JETPRP76
*	MUEFF(200) , XLN(200) , DK(200) , RETURB(200)	JETPRP77
	COMMON /EDGE/ YJETE , SEEDGE	JETPRP78
C*		JETPRP79
	COMMON /MIXER/ MIX,RD(100),XD(100),CF,YR(100)	JETPRP80
	LOGICAL MIX	JETPRP81
	COMMON /GLOBAL/ MAXIT,SUPB,NIT,PSID,YDD,YDC,	JETPRP82
*	P1,P2,UCL,TOL,UPSTRM,CVG	JETPRP83
	LOGICAL SUPB,CVG,UPSTRM	JETPRP84
	COMMON /ACONVG/ YC(100),PD(100),INDC(100), CHOKE, CHOKED	JETPRP85
	LOGICAL CHOKE, CHOKED	JETPRP86
	COMMON /FIT/ CLSP(100)	JETPRP87
	COMMON /STA2/ MACH2,TS2,SS2,V2,RHO2,NPDX2	JETPRP88
	REAL MACH2	JETPRP89
	COMMON /BCMIX2/ GRADU,TW,MUM,RHOM,PIF,ITE	JETPRP90
	REAL MUM	JETPRP91
	COMMON /THRST/WV(100)	JETPRP92
	COMMON /MIXPRP/ MA2(100),VF2(100),TE2(100),TWC(100)	JETPRP93
	REAL MA2	JETPRP94
	COMMON /THERM/ GMC(200),CP(200)	JETPRP95
	COMMON /OUTMIX/ NXORIG	JETPRP96
	COMMON /CHODY/ YCB(100),CLSPCB(100),YCB1 , UCL1	JETPRP97
	COMMON /JET3/ STADD, NV, STATF	JETPRP98
	COMMON /INNAME/ NR, TAH(5) , ND , TAH(4)	JETPRP99
	COMMON /CONSTF/ CON1	JETPRP00

LOGICAL STADD,STATF	JETPRP01
COMMON /SCALD/ SCLD,ALXLM	JETPRP02
LOGICAL SCLD	JETPRP03
COMMON /MOLES/ ALX(100,12)	JETPRP04
COMMON /DIFEQI/	JETPRP05
* NC , CNAME(12) , ALJ(12) , ALJC(12) , ALE(12) ,	JETPRP06
* SCH(12) , TCPRF(12) , HCPRF(12) , CPC(3,12)	JETPRP07
COMMON /YOFXI / IE	JETPRP08
DIMENSION Y(200) , STOR1(200)	JETPRP09
DIMENSION Y(200)	JETPRP10
DIMENSION TKF(200),STOR2(200)	JETPRP11
EQUIVALENCE (TKE(1),DSTOR(601)),(STOR2(1),DSTOR(601))	JETPRP12
DIMENSION YMACH(200)	JETPRP13
EQUIVALENCE (YMACH(1),DSTOR(601))	JETPRP14
EQUIVALENCE (Y(1),DSTOR(401))	JETPRP15
EQUIVALENCE (T(1),DSTOR(1)), (STOR1(1), DSTOR(201))	JETPRP16
EQUIVALENCE (C6,COEF(6))	JETPRP17
EQUIVALENCE (ITWO,TWO)	JETPRP18
DATA NREG/1/	JETPRP19
C* PROPERTIES DIMENSIONLESS UNLESS OTHERWISE NOTED	JETPRP20
C*	JETPRP21
1 IF(.NOT.FENTRY) GO TO 10	JETPRP22
ITWOJ=ITWO,1	JETPRP23
NX=1	JETPRP24
NV=1	JETPRP25
DT0=TJET-TF	JETPRP26
XJ=X(1)	JETPRP27
IGD=1	JETPRP28
C* INITIALIZE AT 1-ST STATION	JETPRP29
C*	JETPRP30
2 UC(1)=UD(1)	JETPRP31
TC(1)=THD(1)	JETPRP32
TIC(1)=TJFT	JETPRP33
PTC(1)=1	JETPRP34
TTC(1)=1	JETPRP35
IF(AMBT0) TTC(1)=BITS	JETPRP36
IF(MJET.EQ. 0.) PTC(1)=BITS	JETPRP37
VJ(1)=YD(NPD)	JETPRP38
C* EVALUATE MASS FLOW IN JET--STREAM FUNCTION	JETPRP39
C*	JETPRP40
3 EXP=1.	JETPRP41
IF(AXI) EXP=2.	JETPRP42
CON1=VJET*C6**EXP	JETPRP43
REFL=B(1)*C6	JETPRP44
99 CALL FMPYC(1,C6,YD,Y,NPD)	JETPRP45
CALL FMPYC(1,EJET,ED,TKE,NPD)	JETPRP46
CALL FMPYC(1,TJET,THD,T,NPD)	JETPRP47
IF(SCLD) GO TO 98	JETPRP48
CALL SCALE(UD,ITWOJ,1,X(1))	JETPRP49
GO TO 97	JETPRP50

96	CALL SCALE(ALX(1,NC),ITWOJ,1,X)	JETPRP51
97	CALL PROPJ(ITWOJ,TURBJ,1,XJ,Y,T,KE,i,NPD)	JETPRP52
	CALL GAMCP(T,GMC,CP,RG,1,NPD)	JETPRP53
	DO 5 L=1,NPD	JETPRP54
	RAD=1.	JETPRP55
	IF(AXI) RAD=YD(L)	JETPRP56
5	STOR1(L)=.5*RHO(L)*UD(L)*RAD	JETPRP57
6	PSI(1)=0.	JETPRP58
66	CALL INTG(STOR1,YD,PSI,2,NPD)	JETPRP59
C*		JETPRP60
C*	IF MIXFR NOZZLE CASE, COMPUTE STREAM	JETPRP61
C*	FUNCTION COINCIDENT WITH DUCT WALL	JETPRP62
C*		JETPRP63
	IF(.NOT. MIX) GO TO 666	JETPRP64
	YD=YR(1)	JETPRP65
	YI=YD(NPD)	JETPRP66
	RHOE=YJET*RHO(1)/TE	JETPRP67
	DPSIF=.5*RHOE*VE/(EXP*VJET)*(YD**EXP-YI**EXP)	JETPRP68
	PSID=PSI(NPD)+DPSIE	JETPRP69
	PD(1)=PE	JETPRP70
	MA2(1)=ME	JETPRP71
	VE2(1)=VE	JETPRP72
	TE2(1)=TE	JETPRP73
	YCD(1)=5ITS	JETPRP74
	TWC(1)=8ITS	JETPRP75
666	WJ(1)=1.	JETPRP76
	FLOWJ=PSI(NPD)*CON1	JETPRP77
	CPJ=GAM*RG/((GAM-1.)*GC/GCJ	JETPRP78
	SUBSON=.TRUE.	JETPRP79
	IF(ME.GE.1. .OR. MJET.GE.1.) SUBSON=.FALSE.	JETPRP80
	GO TO 8760	JETPRP81
C*		JETPRP82
C*	ENTRIES AT STATIONS OTHER THAN STATION 1	JETPRP83
C*		JETPRP84
10	UC(NX)=UD(1)	JETPRP85
	TC(NX)=THD(1)	JETPRP86
	TIC(NX)=SQRT(2.*EJET*GCJ*ED(1)/3.)/VJET	JETPRP87
C*		JETPRP88
11	VJ(NX)=YJETE	JETPRP89
	YSEI=YJETE	JETPRP90
	IF(TWO) YSFT=EDGE1	JETPRP91
	IF(NREG.NF.2) NREG=1	JETPRP92
	IF(.NOT. SCLO) GO TO 1110	JETPRP93
	IF(ALX(1,NC).GE.ALXLTH) NREG=2	JETPRP94
	GO TO 1111	JETPRP95
1110	IF(UD(2).NE.UD(3)) NREG=2	JETPRP96
1111	GO TO (12,14) , NREG	JETPRP97
C*		JETPRP98
C*	POTENTIAL CORE REGION	JETPRP99
C*		JETPRP00

12	DO 13 L=1,NPD	JETPRP01
	IF(SCLD) GO TO 1200	JETPRP02
	IF(UD(L).NE.UCL1) GO TO 15	JETPRP03
	GO TO 13	JETPRP04
1200	IF(ALX(L,NC),GE.ALXLIM) GO TO 15	JETPRP05
13	CONTINUE	JETPRP06
15	B(NX)=YSET-YD(L=1)	JETPRP07
	GO TO 20	JETPRP08
14	B(NX)=2.*Y.IETE	JETPRP09
C*		JETPRP10
C*	CL TOTAL PRESSURE	JETPRP11
C*		JETPRP12
C*		JETPRP13
20	IF(.NOT. CMPS) GO TO 22	JETPRP14
	TEMP=THD(1)*TJET	JETPRP15
	VJC=VJET*UD(1)	JETPRP16
	GAM=GMC(1)	JETPRP17
	CPJ=CP(1)	JETPRP18
	GAM=GAMH(T1)	JETPRP19
	TTCL=TEMP+.5*VJC*VJC/(GCJ*CPJ)+EJET*ED(1)/CPJ	JETPRP20
	IF(.NOT. AMBTG) GO TO 1337	JETPRP21
	TTC(NX)=RITS	JETPRP22
	GO TO 1338	JETPRP23
1337	TTC(NX)=(TTCL-TE)/(TTO-TE)	JETPRP24
1338	CONTINUE	JETPRP25
	XMJ=VJC/SQRT(GAM*GC*RG*TEMP)	JETPRP26
	TTRJ=TTCL/TEMP	JETPRP27
	PK=PE	JETPRP28
	IF(MTX) PK=P2	JETPRP29
	POC=PK*(1+.5*(GAM-1.)*XMJ**2)**(GAM/(GAM-1.))	JETPRP30
	PTC(NX)=(POC-PE)/(PTJET-PE)	JETPRP31
C*		JETPRP32
C*	LOCATE SONIC LINE IF SUPERSONIC***	JETPRP33
C*		JETPRP34
8760	IF(SURSON) GO TO 29	JETPRP35
	TMACH=.FALSE.	JETPRP36
	SUPERT=.TRUE.	JETPRP37
	LK=2	JETPRP38
	IE = 1	JETPRP39
	DO 211 L=1,NPD	JETPRP40
	IF(L.GT.1 .AND. UD(L).EQ. UCL1) LK=L	JETPRP41
	TL=THD(L)*TJET	JETPRP42
	GAM=GAMH(T1)	JETPRP43
	VSON=SQRT(GAM*GC*RG*TL)	JETPRP44
	YMACH(L)=UD(L)*VJET/VSON	JETPRP45
	IF(L.LT.NPD .AND. YMACH(L).EQ. YMACH(L=1)) LK=L	JETPRP46
	IF(.NOT. SUPERT) GO TO 217	JETPRP47
	SUPERT= YMACH(L).LT. 1.	JETPRP48
217	IF(TMACH) GO TO 211	JETPRP49
	TMACH=YMACH(L).GE.1.	JETPRP50

211 CONTINUE	JETPRP51
IF(SUPERT) GO TO 29	JETPRP52
IF(.NOT. TMACH) SUBSON=.TRUE.	JETPRP53
212 IF(.NOT. SUBSON) GO TO 213	JETPRP54
YSONTC(NX)=0.	JETPRP55
GO TO 30	JETPRP56
213 YSONTC(NX)=YOF(1.,YMACH,YD,LK,NPD=1)	JETPRP57
IF(.NOT. ERR) GO TO 30	JETPRP58
ERR=.FALSE.	JETPRP59
29 YSONTC(NX)=BIT9	JETPRP60
GO TO 30	JETPRP61
22 PTC(NX)=BIT9	JETPRP62
TTC(NX)=BIT8	JETPRP63
C*	JETPRP64
C*	JETPRP65
C* EVALUATE MASS FLOW IN JET	JETPRP66
C*	JETPRP67
30 IF(NX.EQ.1) GO TO 1000	JETPRP68
WJ(NX)=SFEDGE/FLOWJ*CON1	JETPRP69
C*	JETPRP70
C*	JETPRP71
C* STORE PARAMETERS FOR CONFINED JET	JETPRP72
C*	JETPRP73
1000 IF(.NOT. MIX) GO TO 1001	JETPRP74
IF(STADD) GO TO 1001	JETPRP75
IF(NX.EQ.1) GO TO 946	JETPRP76
YCD(NV)=YDC	JETPRP77
PD(NV)=P2	JETPRP78
IF(.NOT. SUBSON) INDC(NV)=1	JETPRP79
IF(CHOKE) INDC(NV)=2	JETPRP80
MA2(NV)=MACH2	JETPRP81
VE2(NV)=V2	JETPRP82
TE2(NV)=TS2	JETPRP83
C*	JETPRP84
C* IF MIXER NOZZLE CASE, CALCULATE THRUST	JETPRP85
C*	JETPRP86
946 CALL THRUST(NV)	JETPRP87
C*	JETPRP88
C* IF FLOW CHOKED, CHECK AVAILABLE AREA AT NEXT STATION.	JETPRP89
C* IF .LE. CURRENT AREA TERMINATE	JETPRP90
C*	JETPRP91
IF(.NOT. CHOKE) GO TO 1001	JETPRP92
A1=YDD**EXP-YCB1**EXP	JETPRP93
IF(NV.EQ. NXORIG) GO TO 1001	JETPRP94
NV1=NV+1	JETPRP95
XNX=XD(NV1)	JETPRP96
CALL LCFIT(XD,YR,NXORIG,0,XNX,YDNX,1,0,CLSP)	JETPRP97
CALL LCFIT(XD,YCR,NXORIG,0,XNX,YCRNX,1,0,C1SPCR)	JETPRP98
A2=YDNX**EXP-YCB3NX**EXP	JETPRP99
IF(A2.LE. A1) STATF=.TRUE.	JETPRP00

1001 CALL DTEST
RETURN
END

JETPRP01
JETPRP02
JETPRP03

CJMESHM	MESH REDISTRIBUTION FOR TRANS. REGION	JMESHM01
SUBROUTINE JMESHM		JMESHM02
COMMON /PARAM/ DUM(1601),SM(200),NM,DUM1(7)		JMESHM03
COMMON /CIRL/ DUM2(15), NPD, DUM3(80)		JMESHM04
COMMON /UMFSH/ MCHANG,CK,DY1,NMSH	, CXPC,CXTP,NRED	JMESHM05
C*		JMESHM06
C*		JMESHM07
1 PSIE=SM(NPD)		JMESHM08
DPSIF=DY1*PSIE		JMESHM09
CK1=CK-1.		JMESHM10
CON=DPSIF/CK1		JMESHM11
DO 5 L=1,NMSH		JMESHM12
EXP=FLOAT(1-1)		JMESHM13
5 SM(L)=CON*(CK**EXP-1.)		JMESHM14
NPD=NMSH		JMESHM15
10 RETURN		JMESHM16
END		JMESHM17

CJTCTRL	MAIN JET CONTROL ROUTINE	JTCTRL01
	SUBROUTINE JTCTRL	JTCTRL02
	LOGICAL SUPC,SUPSTP	JTCTRL03
	LOGICAL MCHANG	JTCTRL04
	LOGICAL CURSTP	JTCTRL05
	LOGICAL BYPASS	JTCTRL06
	LOGICAL EOF , ERR	JTCTRL07
	LOGICAL AXI , XPRN , CMPRS , OJET , TURBJ , CORE	JTCTRL08
	REAL MJET , ME , MUREF	JTCTRL09
	COMMON /RSTART/ NREG,RESTR,NRES,MIXPRE	JTCTRL10
	LOGICAL MIXPRE	JTCTRL11
C*		JTCTRL12
	COMMON /JET TWO/	JTCTRL13
	* TWO , DIAO , MJETO , TJETO , VJETO ,	JTCTRL14
	* PTJETO , TIJETO , NJO	JTCTRL15
	REAL MJETO,MACHO	JTCTRL16
	COMMON /BCO/ UO, EO, THO	JTCTRL17
	COMMON /CTRL2/	JTCTRL18
	* EDGEI , SFI , MERGE , XMERGE , YMERGE ,	JTCTRL19
	* SLOPEI , SLOPED , CEPTI , CEPTO	JTCTRL20
	COMMON /MERGET/ MER, MERSTP , XMRG	JTCTRL21
	LOGICAL TWO, MERGE , MER , MERSTP	JTCTRL22
C*****	INPUT COMMON	JTCTRL23
C*		JTCTRL24
	COMMON /INPJET/	JTCTRL25
	* DIAJ , MJET , TJET , PTJET , VJET ,	JTCTRL26
	* TIJET ,	JTCTRL27
	* PE , VE , ME , TIE , TE ,	JTCTRL28
	* AXI , NJ , NMAX ,	JTCTRL29
	* X(100) , XPRN(100) ,	JTCTRL30
	* GAM , RG , PR , PRT ,	JTCTRL31
	* SC , TREF , MUREF	JTCTRL32
C*		JTCTRL33
C*****	CONTROL COMMON	JTCTRL34
C*		JTCTRL35
	COMMON /CTRL/	JTCTRL36
	* NXTA , CMPRS , OJET , TURBJ , COEF(10) ,	JTCTRL37
	* NPU , NPD , DXC , XU , XD ,	JTCTRL38
	* DSTOR(800)	JTCTRL39
C*		JTCTRL40
C*****	PROFILE COMMON	JTCTRL41
C*		JTCTRL42
	COMMON /PROF/ PSI(200),Y(200),UD(200),THD(200),EO(200)	JTCTRL43
C*		JTCTRL44
C*****	CONSTANT AND ERROR COMMON	JTCTRL45
C*		JTCTRL46
	COMMON /CNERR/ BITS , ERR , GC , GCJ , FOOT	JTCTRL47
C*		JTCTRL48
C*****	BOUNDARY CONDITION COMMON	JTCTRL49
C*		JTCTRL50

COMMON /BC/ UEDGE , EEDGE , THEDGE	JTCTRL51
C*	JTCTRL52
C***** POTENTIAL CORE COMMON	JTCTRL53
C*	JTCTRL54
COMMON /CORED/ XCORE , CORE , CORSTP	JTCTRL55
COMMON /SUPER/ SUPC,SUPSTP,XSUP	JTCTRL56
C*	JTCTRL57
C***** SCALER (UNITS CONVERSION) COMMON	JTCTRL58
C*	JTCTRL59
COMMON /SCALER/ SP , SV , SLEN	JTCTRL60
C*	JTCTRL61
C***** JET PROPERTIES COMMON	JTCTRL62
C*	JTCTRL63
COMMON /JET/	JTCTRL64
* B(100) , UC(100) , TC(100) , TIC(100) ,	JTCTRL65
* PTC(100) , WJ(100) , YJ(100) ,	JTCTRL66
* YSONIC(100)	JTCTRL67
COMMON /JET1/ FLOWJ, IT0, NXN , EJET	JTCTRL68
COMMON /JET2/ TTC(100)	JTCTRL69
COMMON /UMESH/ MCHANG,CK,DY1,NM5H , CXPC,CXTP,NRED	JTCTRL70
COMMON /JET3/ STADD, NV, STATF	JTCTRL71
LOGICAL STADD,STATF	JTCTRL72
COMMON /MIXER/ MIX,RD(100),ZD(100),CF,YR(100)	JTCTRL73
LOGICAL MIX	JTCTRL74
DATA BYPASS/F/	JTCTRL75
C*	JTCTRL76
C* INITIALIZE AT FIRST STATION--INCLUDE STATION WHERE CORE	JTCTRL77
C* DISAPPEARS AS A CALCULATION STATION TO BE INSERTED	JTCTRL78
C*	JTCTRL79
NCALC=NXTA	JTCTRL80
1 CORE=.FALSE.	JTCTRL81
SUPC=.FALSE.	JTCTRL82
MER=.FALSE.	JTCTRL83
NXP=NRES	JTCTRL84
NXN=NXP+1	JTCTRL85
XU=X(NXP)	JTCTRL86
IF(.NOT. MIX) GO TO 2	JTCTRL87
NV=2	JTCTRL88
STADD=.FALSE.	JTCTRL89
STATF=.FALSE.	JTCTRL90
2 XD=X(NXN)	JTCTRL91
DXC=XD-XU	JTCTRL92
C*	JTCTRL93
C* CALL JTSTEP FOR INTEGRATION TO NEXT CALCULATION STATION	JTCTRL94
C*	JTCTRL95
3 CALL JTSTEP	JTCTRL96
IF(ERR) GO TO 1000	JTCTRL97
C*	JTCTRL98
C* TEST FOR END OF POTENTIAL CORE	JTCTRL99
C*	JTCTRL00

C*		JTCTRL01
C*	ALSO TEST FOR DISAPPEARANCE OF SUPERSONIC CORE IF JET IS SUPERSONIC	JTCTRL02
C*		JTCTRL03
	IF(CORE .AND. CORSTP) GO TO 5	JTCTRL04
	IF(SUPC .AND. SUPSTP) GO TO 5	JTCTRL05
	IF(MER .AND. MERSTP) GO TO 5	JTCTRL06
	GO TO 20	JTCTRL07
C*		JTCTRL08
C*	RELOCATE JET PROPERTIES UP 1 LOCATION--	JTCTRL09
C*	INSERT XDCORE AND SET XPRN T	JTCTRL10
C*		JTCTRL11
	5 ASSIGN 9 TO LGO	JTCTRL12
	XTEST = X(NXN)	JTCTRL13
	IF((XTEST.NE.XCORE).AND.(XTEST.NE.XSUP).AND.(XTEST.NE.XMRG))	JTCTRL14
	* GO TO 501	JTCTRL15
	ASSIGN 15 TO LGO	JTCTRL16
	GO TO 8	JTCTRL17
501	NM = -(NXTA-NXN+1)	JTCTRL18
	N1=NXN+1	JTCTRL19
	CALL MOVE(5,X(NXN),X(N1),NM,1,R(NXN),B(N1),NM,1,	JTCTRL20
	* TTC(NXN),TTC(N1),NM,1,XPRN(NXN),XPRN(N1),NM,1,	JTCTRL21
	* YSDNIC(NXN),YSDNIC(N1),NM,1)	JTCTRL22
	CALL MOVE(5,YJ(NXN),YJ(N1),NM,1,UC(NXN),UC(N1),NM,1,	JTCTRL23
	* TIC(NXN),TIC(N1),NM,1,PTC(NXN),PTC(N1),NM,1,	JTCTRL24
	* TIC(NXN),TIC(N1),NM,1)	JTCTRL25
	CALL MOVE(1,WJ(NXN),WJ(N1),NM,1)	JTCTRL26
	6 IF(CORE) X(NXN)=XCORE	JTCTRL27
	IF(SUPC) X(NXN)=XSUP	JTCTRL28
	IF(MER) X(NXN)=XMRG	JTCTRL29
	7 XPRN(NXN)=.TRUE,	JTCTRL30
	8 IF(CORE) CORE=.FALSE.	JTCTRL31
	IF(SUPC) SUPC=.FALSE.	JTCTRL32
	IF(MER) MER=.FALSE.	JTCTRL33
	GO TO LGO , (9,15)	JTCTRL34
	9 NCALC = NCALC+1	JTCTRL35
	NXTA=NXTA+1	JTCTRL36
C*		JTCTRL37
C*	COMPUTE JET PROPERTIES	JTCTRL38
C*		JTCTRL39
	15 IF(MIX) STADD=.TRUE.	JTCTRL40
	20 CALL JETPRP	JTCTRL41
	21 IF(ERR) GO TO 1000	JTCTRL42
C*		JTCTRL43
C*	PRINT PROFILES IF REQUESTED	JTCTRL44
C*		JTCTRL45
	30 IF(XPRN(NXN)) CALL JIOUTP	JTCTRL46
C*		JTCTRL47
C*	IF CORSTP=T, REDISTRIBUTE MESH AFTER DISAPPEARANCE	JTCTRL48
C*	OF POTENTIAL CORE	JTCTRL49
C*		JTCTRL50

40 IF(.NOT. CORSTP) GO TO 50	JTCTRL51
IF(BYPASS) GO TO 50	JTCTRL52
IF(MCHANG) CALL JMESHM	JTCTRL53
BYPASS=.TRUE.	JTCTRL54
C*	JTCTRL55
C* INCREMENT COUNTERS,,, TEST FOR END OF PROBLEM	JTCTRL56
C*	JTCTRL57
50 NXN=NXN+1	JTCTRL58
NXP=NXP+1	JTCTRL59
IF(NXN.GT. NCAIC) GO TO 1000	JTCTRL60
IF(MIX .AND. STAFF) GO TO 1000	JTCTRL61
IF(.NOT. STADD) NV=NV+1	JTCTRL62
IF(MIX) STADD=.FALSE.	JTCTRL63
C*	JTCTRL64
C* CONTINUE INTEGRATION	JTCTRL65
C*	JTCTRL66
100 XU=XD	JTCTRL67
GO TO 2	JTCTRL68
C*	JTCTRL69
1000 RETURN	JTCTRL70
END	JTCTRL71

CJTEDGE	LOCATE EDGE OF JET	NEW Y CO-ORDINATES	JTEDGE01
	SUBROUTINE JTEDGE(X,YE,PSIE,ADDP)		JTEDGE02
	LOGICAL HALF		JTEDGE03
	LOGICAL AXI		JTEDGE04
	COMMON /JETTWO/		JTEDGE05
*	TWO , DIAO , MJETO , TJETO , VJETO ,		JTEDGE06
*	PTJETO , TIJETO , NJO		JTEDGE07
	REAL MJETO,MACHO		JTEDGE08
	COMMON /BCO/ UO, EO, THO		JTEDGE09
	COMMON /CTRL2/		JTEDGE10
*	EDGEI , SFI , MERGE , XMERGE , YMERGE ,		JTEDGE11
*	SLOPEI , SLOPED , CEPTI , CEPTD		JTEDGE12
	COMMON /MERGET/ MER, MERSTP , XMRG		JTEDGE13
	COMMON /PROPJ2/ MACHQ,REFID,YII,YO,MERGP		JTEDGE14
	LOGICAL MERGP		JTEDGE15
	LOGICAL TWO, MERGE , MER , MERSTP		JTEDGE16
	COMMON /MISC/ PM(10)		JTEDGE17
	LOGICAL ADDP		JTEDGE18
	COMMON /SETNEW/ LKK,LCDR		JTEDGE19
	COMMON /PRDF/ PSI(200),Y(200),UD(200),THD(200),EO(200)		JTEDGE20
	COMMON /YOFXI/ LEI		JTEDGE21
	COMMON /TNPJET/ DIAJ,MJET,TJET,DUMI1(7),TE,AXI,DUMI2(203),RG		JTEDGE22
	COMMON /RC/ UEDGE,EEDGE,THEDGE		JTEDGE23
	COMMON /PROPJ1/ P,DUMPR(1412)		JTEDGE24
	COMMON /PARAM/ UNIF(200),PSI1(200),DUMP(1201),SM(200),NM,		JTEDGE25
*	DUMP1(7)		JTEDGE26
	COMMON /ERASE/ DUME(800)		JTEDGE27
	COMMON /CTRL/ DUMC1(15), NPD , DUMC2(803)		JTEDGE28
	COMMON /XPRIN/ DPRIN		JTEDGE29
C*			JTEDGE30
C*	AITER1-- COMPUTE B.C. FOR NEXT TRIAL AT MASS BALANCE		JTEDGE31
C*			JTEDGE32
	COMMON /MIXER/ MIX,RD(100),XD(100),CF,YR(100)		JTEDGE33
	LOGICAL MIX		JTEDGE34
	COMMON /GLOBAL/ MAXIT,SUPB,NIT,PSID,YDD,YDC,		JTEDGE35
*	P1,P2,UCL,IQL,UPSTRM,CVG		JTEDGE36
	LOGICAL SUPB,CVG,UPSTRM		JTEDGE37
	COMMON /CNERR/ BITS,ERR,GC,GCJ,FOOT		JTEDGE38
	LOGICAL ERR		JTEDGE39
	COMMON /ACONVG/ YCD(100),PD(100),INDC(100), CHOKE, CHOKED		JTEDGE40
	LOGICAL CHOKE, CHOKED		JTEDGE41
	COMMON /DEIT/ CLSP(100)		JTEDGE42
	COMMON /STA2/ MACH2,TS2,SS2,V2,RHO2,DPOX2		JTEDGE43
	REAL MACH2		JTEDGE44
	COMMON /BCMIX2/ GRADU,TW,MHW,RHGW,PIF,ITE		JTEDGE45
	REAL MHW		JTEDGE46
	COMMON /CHNDY/ YCB(100),CLSPCB(100),YCB1 , UCL1		JTEDGE47
	COMMON /JUMIX/ NXDRI6		JTEDGE48
C*			JTEDGE49
	LOGICAL DPRIN		JTEDGE50

COMMON /SCALD/ SCLD,ALXLIM	JTEDGE51
LOGICAL SCLD	JTEDGE52
COMMON /MOIES / ALX(100,12)	JTEDGE53
COMMON /DIFF01/ NC,DDIF(120)	JTEDGE54
EQUIVALENCE (IE, LE1)	JTEDGE55
DIMENSION STORI(200)	JTEDGE56
EQUIVALENCE (STORI(1),DUMP(1))	JTEDGE57
DIMENSION YI(200),RHO(200)	JTEDGE58
EQUIVALENCE (YI(1),DUMP(201)),(RHO(1),DUMP(401))	JTEDGE59
DATA LFFF/200/	JTEDGE60
DATA IQF/0/	JTEDGE61
C*	JTEDGE62
C*	JTEDGE63
C* IF HALF=.TRUE, EDGE IS TAKEN AT 50 PCT VELOCITY LINE (Y102)	JTEDGE64
*	JTEDGE65
IE = 1	JTEDGE66
ADDP=.FALSE.	JTEDGE67
HALF=.FALSE.	JTEDGE68
IF(PH(4).NE.0.) HALF=.TRUE.	JTEDGE69
C*	JTEDGE70
C* CHECK JET UD FOR 1-ST POINT WHERE UD=UEDGE	JTEDGE71
C* REDEFINE JET EDGE CONDITIONS	JTEDGE72
DO 6620 L=1,NPD	JTEDGE73
IF(UD(L).EQ.UEDGE .AND. UEDGE.EQ. 0.) GO TO 6621	JTEDGE74
6620 CONTINUE	JTEDGE75
6621 NPD=L	JTEDGE76
IF((.NOT. MIX) .OR. (MIX .AND. UPSTRM)) THD(NPD)=THEDGE	JTEDGE77
IF((.NOT. MIX) .OR. (MIX .AND. UPSTRM)) ED(NPD)=EEDGE	JTEDGE78
NPD1=NPD-1	JTEDGE79
DIFEU=ABS(UEDGE-UD(NPD-1))	JTEDGE80
IF(.NOT. MIX) GO TO 1	JTEDGE81
DO 6720 L=1,NPD	JTEDGE82
6720 RHO(L)=144.*P2/(RG*THD(L)*TJET)	JTEDGE83
LE1=NPD	JTEDGE84
LF=LE1-1	JTEDGE85
PSTE=SM(LE1)	JTEDGE86
RHUE=RHO(LE1)	JTEDGE87
UEE=UD(LE1)	JTEDGE88
GO TO 468	JTEDGE89
C*	JTEDGE90
C*	JTEDGE91
C* LOCATE EDGE(S) OF JET(S)	JTEDGE92
C* -- POINT WHERE VELOCITY DIFFERENCE IS .98*MAX. VELOCITY	JTEDGE93
C* DIFFERENCE ACROSS THE JET	JTEDGE94
C*	JTEDGE95
C*	JTEDGE96
1 UCL1=UD(1)	JTEDGE97
LK=1	JTEDGE98
DO 2 L=1,NPD	JTEDGE99
UDIF(L)=ABS(UD(L)-UCL1)	JTEDGE00

IF(L.EQ,1) GO TO 2	JTEDGE01
TESTA=UD(L-1)+1.E-6	JTEDGE02
TESTB=UD(L-1)-1.E-6	JTEDGE03
IF(UD(L).GE.TESTB .AND. UD(L).LE.TESTA .AND. L.LT.LFFF) LK=L	JTEDGE04
2 RHO(L)=144.*P/(RG*THD(L)*TJET)	JTEDGE05
TESTE=AMAX1(.98,PM(3))	JTEDGE06
C	JTEDGE07
DETERMINE EDGE FROM MOLE FRACTION	JTEDGE08
C	JTEDGE09
IF(IOF, EQ, 0) GO TO 2220	JTEDGE10
IF(.NOT. SCLD) GO TO 2220	JTEDGE11
TESTE = AMAX1(.98, PM(3))	JTEDGE12
DO 2208 LE1=1,NPD	JTEDGE13
IF(ALX(LE1,NC).GE.TESTE) GO TO 2209	JTEDGE14
2208 CONTINUE	JTEDGE15
2209 LE = LE1-1	JTEDGE16
SLOPED= (SM(LE1)-SM(LE))/(ALX(LE1,NC)-ALX(LE,NC))	JTEDGE17
PSIE = SM(LE)+SLOPED*(IFSTE-ALX(LE,NC))	JTEDGE18
CALL MOVE(1,SM,PSIE,LE,1)	JTEDGE19
PSIE(LF1)= PSIE	JTEDGE20
DO 2210 L=1,LE	JTEDGE21
2210 STORI(L)= 2./(RHO(L)*UD(L))	JTEDGE22
RHOE = YOF(PSIE,SM,RHO,1,NPD)	JTEDGE23
UEE = YOF(PSIE,SM,UD,1,NPD)	JTEDGE24
STORI(LE1)= 2./(RHOE*UEE)	JTEDGE25
YI(1) = 0.	JTEDGE26
CALL INTG(STORI,PSIE,YI,2,LE1)	JTEDGE27
YE = YI(LE1)	JTEDGE28
YSV = YE	JTEDGE29
IF(AXI) YE=SQRT(2.*YE)	JTEDGE30
CALL MOVE(1,YI,Y,LE,1)	JTEDGE31
CALL MOVE(1,ALX(1,NC),UDIF,NPD,1)	JTEDGE32
GO TO 4001	JTEDGE33
2220 ULM = TESTE*ABS(UDGE-UCL1)	JTEDGE34
IF(HALF) ULM=.5*ABS(UDGE-UD(1))	JTEDGE35
LFFF=LK	JTEDGE36
IF(OPRIV) CALL TABPRT(2MLK,LK,1,1,0)	JTEDGE37
IF(OPRIV) CALL TABPRT(4HUDIF,UDIF,NPD,10,0)	JTEDGE38
C*	JTEDGE39
C*	JTEDGE40
C**** LOCATE OUTER STREAMLINE * ****	JTEDGE41
C*	JTEDGE42
C* FOR OUTER STREAMLINE-- CHOOSE THE MINIMUM OF INTERPOLATED	JTEDGE43
C* AND EXTRAPOLATED VALUES	JTEDGE44
C*	JTEDGE45
LE1=0	JTEDGE46
L = NPD+1	JTEDGE47
300 L = L-1	JTEDGE48
IF(UDIF(L).LT.ULM) GO TO 301	JTEDGE49
IF(L.GE.LK) GO TO 300	JTEDGE50

301 LE1 = L+1	JTEDGE51
LF=LE1-1	JTEDGE52
SLOPED=(SM(LE1)-SM(LE))/(UDIF(LE1)-UDIF(LE))	JTEDGE53
PSIE1=SM(LF)+SLOPED*(ULIM-UDIF(LE))	JTEDGE54
IF(.NOT. HALF) GO TO 6	JTEDGE55
PSIE=PSIE1	JTEDGE56
LEE = LE	JTEDGE57
GO TO 8	JTEDGE58
6 LEE=LE1-2	JTEDGE59
SLOPED=(SM(LE)-SM(LEE))/(UDIF(LE)-UDIF(LEE))	JTEDGE60
PSIE2=SM(LF)+SLOPED*(ULIM-UDIF(LE))	JTEDGE61
PSIE=AMIN1(PSIE1,PSIE2)	JTEDGE62
8 UEF = YOF(PSIE,SM,UD,LEE,NPD)	JTEDGE63
RHOE=YOF(PSIE,SM,RHO,LJ,NPD)	JTEDGE64
468 DO 4 L=1,LF	JTEDGE65
4 STORI(L)=2./(RHO(L)*UD(L))	JTEDGE66
STORI(LE1)=2./(RHOE*UEF)	JTEDGE67
C*	JTEDGE68
C*	JTEDGE69
C* MOVE PSI ARRAY TO SCRATCH--STORAGE	JTEDGE70
C*	JTEDGE71
CALL MOVE(1,SM,PSI1,LE,1)	JTEDGE72
C*	JTEDGE73
C* INSERT EDGE STREAMLINE	JTEDGE74
C*	JTEDGE75
PSI1(LE1)=PSIE	JTEDGE76
C*	JTEDGE77
C* DETERMINE Y- COORDINATES BY INTEGRATION OF STREAM FUNCTION	JTEDGE78
C*	JTEDGE79
C*	JTEDGE80
C* USE TRAPEZOIDAL RULE	JTEDGE81
C*	JTEDGE82
YI(1)=0.	JTEDGE83
YCB1=0.	JTEDGE84
IF(.NOT. MIX) GO TO 399	JTEDGE85
CALL LCFIT(XD,YCB,NXORIG,0,X*DIAJ,YI(1),1,0,CLSPCB)	JTEDGE86
YCB1=YI(1)	JTEDGE87
IF(AXI) YI(1)=0.	JTEDGE88
399 CALL INTG(STORI,PSI1,YI,2,LE1)	JTEDGE89
YE=YI(LE1)	JTEDGE90
YSV=YE	JTEDGE91
IF(AXI) YE=SQR1(YCB1**2+2.*YF)	JTEDGE92
C* MOVF YI VALUES TO Y ARRAY	JTEDGE93
C*	JTEDGE94
LX=LF	JTEDGE95
IF(MIX) LX=LF1	JTEDGE96
40 CALL MOVE(1,YI,Y,LX,1)	JTEDGE97
IF(MIX) GO TO 8888	JTEDGE98
C*	JTEDGE99
C* CONTINUE INTEGRATION TO NPD-1	JTEDGE00

C* EXTRAPOLATE FOR NPD POINT	JTEDGE01
C*	JTEDGE02
4001 NPD1 = NPD-1	JTEDGE03
MX=1	JTEDGE04
IF(NPD1.EQ. LE) MX=2	JTEDGE05
IF(NPD1.EQ. LE) GO TO 887	JTEDGE06
DO 42 L=LE1,NPD1	JTEDGE07
42 STORI(L)=2./(RHO(L)*UD(L))	JTEDGE08
CALL INTG(STORI,SM,Y,LE1,NPD1)	JTEDGE09
887 IF(UEDGE.GT. 0.) GO TO 8887	JTEDGE10
GO TO (888,889), MX	JTEDGE11
C 888 SLOPE=(Y(NPD1)-Y(NPD1-1))/(UDIF(NPD1)-UDIF(NPD1-1))	JTEDGE12
C Y(NPD)=Y(NPD1)+SLOPE*(UDIF(NPD)-UDIF(NPD1))	JTEDGE13
888 SLOPE=(Y(NPD1)-Y(NPD1-1))/(SM(NPD1)-SM(NPD1-1))	JTEDGE14
Y(NPD)=Y(NPD1)+SLOPE*(SM(NPD)-SM(NPD1))	JTEDGE15
GO TO 8888	JTEDGE16
C 889 SLOPE=(YSV-Y(NPD1))/(ULIM-UDIF(NPD1))	JTEDGE17
C Y(NPD)=YSV+SLOPE*(UDIF(NPD)-ULIM)	JTEDGE18
889 SLOPE=(YSV-Y(NPD1))/(PSIF-SM(NPD1))	JTEDGE19
Y(NPD)=YSV+SLOPE*(SM(NPD)-PSIE)	JTEDGE20
GO TO 8888	JTEDGE21
8887 STORI(NPD)=2./(RHO(NPD)*UD(NPD))	JTEDGE22
CALL INTG(STORI,SM,Y,NPD,NPD)	JTEDGE23
8888 Y(1)=YCB1	JTEDGE24
IF(.NOT. AXI) GO TO 3811	JTEDGE25
IF(.NOT. DPRIN) GO TO 19	JTEDGE26
CALL TABPRT(4HPSI1,PSI1,LE,10,0)	JTEDGE27
CALL TABPRT(4HY(L),Y,NPD,10,0)	JTEDGE28
CALL TABPRT(5HYI(L),YT,LE,10,0)	JTEDGE29
CALL TABPRT(5HSTORI,STORI,LE,10,0)	JTEDGE30
WRITE(6,107)PM(4),UEDGE,ULIM,PSIE,NPD,NPD1,LE1,LE,MX,HALF,MIX,YCB1	JTEDGE31
107 FORMAT(1X//	JTEDGE32
1 2X,8HPM(4) =,E14.6,2X,8HUEDGE =,E14.6,2X,8HULIM =,E14.6,2X,	JTEDGE33
2 8HPSIE =,E14.6//	JTEDGE34
3 2X,8HNPD =,16,10X,8HNPD1 =,16,10X,8HLE1 =,16,10X,	JTEDGE35
4 8HLE =,16//	JTEDGE36
5 2X,8HMX =,16,10X,8HHALF =,16,10X,8HMIX =,16,10X,	JTEDGE37
6 8HYCB1 =,E14.6)	JTEDGE38
19 CONTINUE	JTEDGE39
DO 7 L=2,NPD	JTEDGE40
7 Y(L)=SQRT(YCB1**2+2.*Y(L))	JTEDGE41
3811 CONTINUE	JTEDGE42
C*	JTEDGE43
C* DETERMINE IF NEW POINT IS TO BE ADDED	JTEDGE44
C*	JTEDGE45
C*	JTEDGE46
C* IF TWO=1 AND MERGE=F, LOCATE BOUNDARY OF INNER JET	JTEDGE47
C*	JTEDGE48
IF(MERGE.NR. (.NOT. TWO)) GO TO 30	JTEDGE49
ULIM=TESTE*ABS(U0-1.)	JTEDGE50

TEST2=U0+1.E-6	JTEDGE51
TEST1=U0-1.E-6	JTEDGE52
C* LOCATE NEAREST MESH POINT	JTEDGE53
C*	JTEDGE54
DO 200 L=1,NPD	JTEDGE55
IF(JD(L).LE.TEST2 .AND. UD(L).GE.TEST1) UD(L)=U0	JTEDGE56
IF(UD(L).EQ.U0) GO TO 259	JTEDGE57
200 CONTINUE	JTEDGE58
C*	JTEDGE59
C* ASSUME JETS HAVE MERGED-- SET CO-ORDINATES OF MERGE POINT	JTEDGE60
C*	JTEDGE61
MERGE=.TRUE.	JTEDGE62
MERGP=.TRUE.	JTEDGE63
SFI=SM(NMERG)	JTEDGE64
EDGEI=Y(NMERG)	JTEDGE65
XMERGE=X	JTEDGE66
YMERGE=EDGEI	JTEDGE67
C*	JTEDGE68
C* COMPUTE COEFFICIENTS IN LINEAR EQUATIONS CONNECTING THE	JTEDGE69
C* NOZZLE CORNERS WITH MERGE POINT	JTEDGE70
C*	JTEDGE71
CEPT=DIAJ/DIAI	JTEDGE72
SLOPFI=(YMERGE-CEPT)/XMERGE	JTEDGE73
CEPT=1.	JTEDGE74
SLOPEI=(YMERGE-1.)/XMERGE	JTEDGE75
GO TO 30	JTEDGE76
C*	JTEDGE77
C* SET NMERGE TO L+1, STORE INNER EDGE SF AND CO-ORDINATE	JTEDGE78
C*	JTEDGE79
259 NU=L	JTEDGE80
DO 260 L=1,NU	JTEDGE81
IF(UDIF(L).GE.ULIML) GO TO 210	JTEDGE82
260 CONTINUE	JTEDGE83
210 NMERG=L	JTEDGE84
NMRG1=L-1	JTEDGE85
SLOPS=(SM(NMERG)-SM(NMRG1))/(UDIF(NMERG)-UDIF(NMRG1))	JTEDGE86
SFI=SM(NMRG1)+SLOPS*(ULIML-UDIF(NMRG1))	JTEDGE87
SLOPY=(Y(NMERG)-Y(NMRG1))/(SM(NMERG)-SM(NMRG1))	JTEDGE88
EDGEI=Y(NMRG1)+SLOPY*(SFI-SM(NMRG1))	JTEDGE89
30 TESTU=.005	JTEDGE90
IF(PM(2).NE.0.) TESTU=PM(2)	JTEDGE91
IF(LKK.LE.3 .AND. UDIF(NPD).EQ.UDIF(NPD1)) LKK=4	JTEDGE92
IF(DIFFU.GT. TESTU .OR. LKK.LE.3) ADDP=.TRUE.	JTEDGE93
LKK=LKK+1	JTEDGE94
IF(OPRIN) WRITE (6,7222) ADDP,DIFFU,TESTU	JTEDGE95
7222 FORMAT(/,12X,L6,2X,2E16.8)	JTEDGE96
C*	JTEDGE97
100 RETURN	JTEDGE98
END	JTEDGE99
	JTEDGE00

CJTFILE	STORAGE OF AERO-DATA FILE FOR PLOT OR NOISE	JTFILE01
	SUBROUTINE JTFILE(NTRY,XX)	JTFILE02
	LOGICAL ENTRYI	JTFILE03
	LOGICAL FOUND	JTFILE04
	LOGICAL MCHANG	JTFILE05
	LOGICAL AMBTO	JTFILE06
	LOGICAL DPRIN	JTFILE07
	LOGICAL EOF , ERR	JTFILE08
	LOGICAL AXI , XPRN , CMPS , QJET , TURBJ , CORE	JTFILE09
	LOGICAL ENTRYI	JTFILE10
	REAL MJET , ME , MUREF	JTFILE11
	REAL MACH	JTFILE12
	REAL MOLF1(100),MOLF2(100),MOLF3(100),MOLF4(100),	JTFILE13
	* MOLF5(100),MOLF6(100),MOLF7(100),MOLF8(100),	JTFILE14
	* MOLF9(100),MOLF10(100),MOLF11(100),MOLF12(100)	JTFILE15
	COMMON /CNFILE/ NF	JTFILE16
	COMMON /RSTART/ NREG,RESTR,NRES,MIXPRE	JTFILE17
	LOGICAL MIXPRE	JTFILE18
	COMMON /CTRL2/ TWODY(9)	JTFILE19
	COMMON /DIFEQ1/	JTFILE20
	* NC,CNAME(12),ALJ(12),ALJO(12),ALE(12),	JTFILE21
	* SCY(12),T,CPHF(12),HCPHF(12),CPC(3,12)	JTFILE22
	COMMON /DICTRL/ DIFF , CND(10)	JTFILE23
	LOGICAL DIFF	JTFILE24
	COMMON /MOLES / ALX(1200)	JTFILE25
	COMMON /RCMDL / ALEDGE(12),ALD(12)	JTFILE26
	COMMON /FLIK/ CSC	JTFILE27
	COMMON /PARAM/	JTFILE28
	* U(200),T(200),TOT(200),XMACH(200),PTOT(200),ITD(200),	JTFILE29
	* PID(200),DUMP9(409)	JTFILE30
	COMMON /RATIO/ AMBTO	JTFILE31
	COMMON /MISC/ P4(10), PLOT	JTFILE32
	COMMON /INPI/ ENTRYI	JTFILE33
	COMMON /UMESH/ MCHANG,CK,DY1,NMSH , CXPC,CXIP,NRED	JTFILE34
C*		JTFILE35
C*****	INPUT COMMON	JTFILE36
C*		JTFILE37
	COMMON /INPJET/	JTFILE38
	* DIAJ , MJET , TJET , PTJET , VJET ,	JTFILE39
	* TIJFT ,	JTFILE40
	* PE , VE , ME , TIE , TE ,	JTFILE41
	* AXI , NJ , NM ,	JTFILE42
	* X(100) , XPRN(100) ,	JTFILE43
	* GAM , RG , PR , PRT ,	JTFILE44
	* SC , TREF , MUREF	JTFILE45
C*		JTFILE46
C*****	CONSTANT AND ERROR COMMON	JTFILE47
C*		JTFILE48
	COMMON /CNERR/ BITS , ERR , GC , GCJ , FOOT	JTFILE49
C*		JTFILE50

C***** BOUNDARY CONDITION COMMON	JTFILE51
C*	JTFILE52
COMMON /BC/ UEDGE , EEDGE , THEDGE	JTFILE53
C*	JTFILE54
C***** POTENTIAL CORE COMMON	JTFILE55
C*	JTFILE56
COMMON /CORED/ XCORE , CORE , CORSTP	JTFILE57
C*	JTFILE58
C***** SCALER (UNITS CONVERSION) COMMON	JTFILE59
C*	JTFILE60
COMMON /SCALER/ SP , SV , SLEN	JTFILE61
C*	JTFILE62
COMMON /JET1/ FLOWJ,T10,NX,EJET	JTFILE63
COMMON /PROPIJ/	JTFILE64
* P , PRL , PRTT , RGAS , SCC ,	JTFILE65
* TRFF , VSREF , MACH , XLC ,	JTFILE66
* REFL,C,CHI,RNORM,	JTFILE67
* RHO(200),DUM13(600),XLN(200),DUM14(400)	JTFILE68
COMMON /XPRIN/ DPRIN	JTFILE69
COMMON /FOATA/ DUM7(4),NFCSV,NRW	JTFILE70
COMMON /CPRJP/ CT1,CT2,CT3,CT4,CT5,CT6,CT7,CT8 ,CT9,CT10	JTFILE71
COMMON/ PRNF/ PSI(200),Y(200),UD(200),THD(200),ED(200)	JTFILE72
COMMON /JET/	JTFILE73
* B(100),JC(100),TC(100),TIC(100),	JTFILE74
* PTC(100),WJ(100),YJ(100),YSDNTIC(100)	JTFILE75
COMMON /JET2/ TTC(100)	JTFILE76
COMMON /CTRL/	JTFILE77
* NXTA , CMPRS , QJET , TURBJ , COEF(10) ,	JTFILE78
* NPU , NPD , DXC , XU , XDD ,	JTFILE79
* TID(200) , DSTOR(600)	JTFILE80
COMMON /TAG/	JTFILE81
* NAME(10),TITLE1(10),IDENT(10),ADDRES(10),IDENT1(10)	JTFILE82
C*	JTFILE83
COMMON /MIXER/ MIX,RD(100),XD(100),CF,YR(100)	JTFILE84
LOGICAL MTX	JTFILE85
COMMON /FLOBAI/ MAXIT,SUPB,NIT,PSID,YDD,YDC,	JTFILE86
* P1,P2,UCL,TOL,UPSTRM,CVG	JTFILE87
LOGICAL SUPB,CVG,UPSTRM	JTFILE88
COMMON /ACNVG/ YC(100),PD(100),INDC(100), CHUKE, CHOKED	JTFILE89
LOGICAL CHUKE, CHOKED	JTFILE90
COMMON /OPT1/ CLSP(100)	JTFILE91
COMMON /STA2/ MACH2,TS2,SS2,V2,RHO2,DPDX2	JTFILE92
REAL MACH2	JTFILE93
COMMON /BGMIX2/ GRADJ,TW,MUM,RHOW,PTF,TTE	JTFILE94
REAL MUM	JTFILE95
COMMON /MIXPRP/ MA2(100),VE2(100),IE2(100),IWC(100)	JTFILE96
REAL MA2	JTFILE97
COMMON /THRST/ WV(100)	JTFILE98
COMMON /OUTMIX/ NXDRIG	JTFILE99
COMMON /CBODY/ YCB(100),CLSPCB(100),YCB1 , UCL1	JTFILE00

COMMON /JET3/STADD,NV,STATF	JTFILE01
COMMON /KEYS/ KEY(11),KEY8(11),KOD8(11),KOD8(11)	JTFILE02
COMMON /MERGE1/ MER,MERSTP,XMR0	JTFILE03
COMMON /SUPER/ SUPC,SUPSTP,XSUP	JTFILE04
COMMON /CPRJP2/ CTP,CTS,CTM	JTFILE05
COMMON /PRJP2/ MACHD,REFLO,YI,YO,MERGE	JTFILE06
COMMON /CONSTF/ CON1	JTFILE07
COMMON /FILIND/ KREC,KXX	JTFILE08
C* EQUIVALENCE (MOLF1,ALX(1)),(MOLF2,ALX(101)),(MOLF3,ALX(201)),	JTFILE09
* (MOLF4,ALX(301)),(MOLF5,ALX(401)),	JTFILE10
* (MOLF6,ALX(501)),(MOLF7,ALX(601)),	JTFILE11
* (MOLF8,ALX(701)),(MOLF9,ALX(801)),	JTFILE12
* (MOLF10,ALX(901)),(MOLF11,ALX(1001)),	JTFILE13
* (MOLF12,ALX(1101))	JTFILE14
C* DIMENSION GJET(200)	JTFILE15
DIMENSION LEN2(11),NS2A(10),LEN1(5),NSA(4)	JTFILE16
EQUIVALENCE (UE,VE)	JTFILE17
EQUIVALENCE (CA,COEF(6))	JTFILE18
DATA ENTRYI/T/	JTFILE19
DATA NSA/15,1HJ,6,1HJ/	JTFILE20
DATA NS2A/5,PHI,1,2HIL,11,1HI,10,2HIJ,5,2HJK/	JTFILE21
DATA IRLANK/0202020202020/	JTFILE22
DATA IPRDF/5HAPROF/	JTFILE23
DATA KXX1/2HXX/	JTFILE24
C* JTFILE25	
C* JTFILE26	
C* JTFILE27	
C* JTFILE28	
C* NTRY=1 WRITE PROFILES AT STATION XX	JTFILE29
C* NTRY=2 WRITE CL PROPS. AND INPUT VARIABLES	JTFILE30
C* NTRY=3 READ CL PROPS AND INPUT VARIABLES	JTFILE31
C* NTRY=4 RECLAIM PROFILE AT STATION XX	JTFILE32
C* JTFILE33	
1 GO TO (10,100,120,200) , NTRY	JTFILE34
C* JTFILE35	
10 IF(.NOT.ENTRYI) GO TO 11	JTFILE36
CALL FMPYC(1,VJET,UD,U,NPD)	JTFILE37
CALL FMPYC(1,TJET,THD,T,NPD)	JTFILE38
KREC = 0	JTFILE39
11 ENTRYI=.FALSE.	JTFILE40
KREC = KREC+1	JTFILE41
KXX = CSC*KXX+.5	JTFILE42
WRITE (3) KREC,KXX,KREC,	JTFILE43
* SUPD,SUPSTP,CORF,CORSTP,MER,MERSTP,NPD,	JTFILE44
* PSI,Y,UD,THD,ED,TID,RMO,XLN,U,I,TOT,XMACH,	JTFILE45
* PIDT,TID,PID,	JTFILE46
* MOLF1,MOLF2,MOLF3,MOLF4,MOLF5,MOLF6,	JTFILE47
* MOLF7,MOLF8,MOLF9,MOLF10,MOLF11,MOLF12,J	JTFILE48
GO TO 1000	JTFILE49
C	JTFILE50

100 REWIND 3

WRITE (2) KXX1,KREC,

* NAME,TITLE1,IDENT,ADDRES,IDENT1,TWOBT,BITS,ERK,GC,GCJ,
* FOOT,DIAJ,MJET,TJET,PTJET,VJET,TIJET,EJET,PE,VE,ME,TIE,TE,
* AXI,NJ,NM,UE,MIXPRE,XLC,FLD4J,MERGE,NV,CON1,
* CT1,CT2,CT3,CT4,CT5,CT6,CT7,CT8,CT9,CTP,CTS,CTM,
* GAM,RG,PR,PRT,SC,TREF,MUREF,SP,SV,SLEN,DPRIN,PLOT,C6,
* MIX,CF,MAXIT,TOL,SUPB,
* X,XPRN,B,HC,TC,TIC,PTC,WJ,YJ,TTC,YSONIC,
* YCB,XD,RO,YR,YCD,PD,WV,MA2,VE2,TE2,NXTA,I,
* NC,CNAME,ALJ,ALJD,ALE,SCM,CPC,DIFF,CSC

DO 101 IJ=1,KREC

READ (3) JREC,KXX,KREG,

* SUPD,SUPSTP,CORE,CORSTP,MER,MERSTP,NPD,
* PSI,Y,JD,THD,ED,TID,RHO,XLN,U,T,TOT,XMACH,
* PTOT,TID,PTD,
* MOLF1,MOLF2,MOLF3,MOLF4,MOLF5,MOLF6,
* MOLF7,MOLF8,MOLF9,MOLF10,MOLF11,MOLF12,J
WRITE (2) JREC,KXX,KREG,
* SUPD,SUPSTP,CORE,CORSTP,MER,MERSTP,NPD,
* PSI,Y,JD,THD,ED,TID,RHO,XLN,U,T,TOT,XMACH,
* PTOT,TID,PTD,
* MOLF1,MOLF2,MOLF3,MOLF4,MOLF5,MOLF6,
* MOLF7,MOLF8,MOLF9,MOLF10,MOLF11,MOLF12,J

101 CONTINUE

GO TO 1000

C

120 READ (1)

* DIAJ,MJET,TIJET,VJET,PE,TE,TIE,VE,GFX,RG,PR,PRT,SC,TREF,
* MUREF,DIFF,NC,CNAME,ALE,SCM,CPC,NJ,NM,CT1,CT2,CT3,CT4,CT5,
* CT6,CT7,CT8,GJET,Y,UD,THD,TID,ED,
* MOLF1,MOLF2,MOLF3,MOLF4,MOLF5,MOLF6,
* MOLF7,MOLF8,MOLF9,MOLF10,MOLF11,MOLF12
IF(NE,GE,2) READ (2) KXX1,KREC,
* NAME,TITLE1,IDENT,ADDRES,IDENT1,TWOBT,BITS,ERK,GC,GCJ,
* FOOT,DIAJ,MJET,TJET,PTJET,VJET,TIJET,EJET,PE,VE,ME,TIE,TE,
* AXI,NJ,NM,UE,MIXPRE,XLC,FLD4J,MERGE,NV,CON1,
* CT1,CT2,CT3,CT4,CT5,CT6,CT7,CT8,CT9,CTP,CTS,CTM,
* GAM,RG,PR,PRT,SC,TREF,MUREF,SP,SV,SLEN,DPRIN,PLOT,C6,
* MIX,CF,MAXIT,TOL,SUPB,
* X,XPRN,B,HC,TC,TIC,PTC,WJ,YJ,TTC,YSONIC,
* X,XPRN,B,UC,TIC,PTC,WJ,YJ,TTC,YSONIC,
* YCB,XD,RO,YR,YCD,PD,WV,MA2,VE2,TE2,NXTA,I,
* NC,CNAME,ALJ,ALJD,ALE,SCM,CPC,DIFF,CSC

GO TO 1000

C

200 KHES= CSC*XX+.5

DO 210 IJ=1,KREC

READ (2) JREC,KXX,KREG,

* SUPD,SUPSTP,CORE,CORSTP,MER,MERSTP,NPD,

JTFILE51

JTFILE52

JTFILE53

JTFILE54

JTFILE55

JTFILE56

JTFILE57

JTFILE58

JTFILE59

JTFILE60

JTFILE61

JTFILE62

JTFILE63

JTFILE64

JTFILE65

JTFILE66

JTFILE67

JTFILE68

JTFILE69

JTFILE70

JTFILE71

JTFILE72

JTFILE73

JTFILE74

JTFILE75

JTFILE76

JTFILE77

JTFILE78

JTFILE79

JTFILE80

JTFILE81

JTFILE82

JTFILE83

JTFILE84

JTFILE85

JTFILE86

JTFILE87

JTFILE88

JTFILE89

JTFILE90

JTFILE91

JTFILE92

JTFILE93

JTFILE94

JTFILE95

JTFILE96

JTFILE97

JTFILE98

JTFILE99

JTFILE00

* PSI,Y,UD,THD,ED,TID,RHO,XLN,U,I,TOT,XMACH,	JTFILE01
* PTOT,TTD,PTD,	JTFILE02
* MOLF1,MOLF2,MOLF3,MOLF4,MOLF5,MOLF6,	JTFILE03
* MOLF7,MOLF8,MOLF9,MOLF10,MOLF11,MOLF12,J	JTFILE04
WRITE (3) JREC,KXX,KREG,	JTFILE05
* SUPD,SUPSTP,CORE,CORSTP,MER,MERSTP,NPD,	JTFILE06
* PSI,Y,UD,THD,ED,TID,RHO,XLN,U,I,TOT,XMACH,	JTFILE07
* PTOT,TTD,PTD,	JTFILE08
* MOLF1,MOLF2,MOLF3,MOLF4,MOLF5,MOLF6,	JTFILE09
* MOLF7,MOLF8,MOLF9,MOLF10,MOLF11,MOLF12,J	JTFILE10
IF(KXX.EQ.KRES) GO TO 220	JTFILE11
210 CONTINUE	JTFILE12
ERR = .TRUE.	JTFILE13
WRITE (6,211) XX	JTFILE14
211 FORMAT(1H1,///2X,10HSTATION X=,F10.6,2X,11HNOT ON FILE//)	JTFILE15
GO TO 1000	JTFILE16
220 WRITE (6,221) XX	JTFILE17
221 FORMAT(1H1,///2X,20H*JETMIX RESTART X=,F10.6//)	JTFILE18
IF(MIX) MIXPRE=,TRUE,	JTFILE19
C	JTFILE20
1000 RETURN	JTFILE21
END	JTFILE22

CJTINIT	INITIALIZE NEW PROBLEM	JTINIT01
	SUBROUTINE JTINIT	JTINIT02
C*****		JTINIT03
C*****	SPECIAL VERSION FOR 12 SPECIES	JTINIT04
C*****		JTINIT05
	LOGICAL DPRIN	JTINIT06
	INTEGER PIOT	JTINIT07
	LOGICAL ENRY1	JTINIT08
	LOGICAL MCHANG	JTINIT09
	LOGICAL ERR	JTINIT10
	LOGICAL AXI , XPRN , CMPRS , QJET , TURBJ , CORE	JTINIT11
	REAL MJET , ME , MUREF	JTINIT12
	COMMON /RSTART/ NREG,RESTRT,NRES,MIXPRE	JTINIT13
	LOGICAL MIXPRE	JTINIT14
	COMMON /DIFEQ1/	JTINIT15
	* NC , CNAME(12) , ALJ(12) , ALJD(12) , ALE(12) , SCM(12) ,	JTINIT16
	* TCPPF(12) , HCPPF(12) , CPC(3,12)	JTINIT17
	COMMON /DICTRL/ DIFF , CND(10)	JTINIT18
	LOGICAL DIFF	JTINIT19
	COMMON /MOLES / ALX(100,12)	JTINIT20
	COMMON /BCMDL / ALFDGE(12),ALD(12)	JTINIT21
C*		JTINIT22
C*****	INPUT COMMON	JTINIT23
C*		JTINIT24
	COMMON /INPJET/	JTINIT25
	* DIAJ , MJET , TJET , PIJET , VJET ,	JTINIT26
	* TIJET ,	JTINIT27
	* PE , VE , ME , TIE , TF ,	JTINIT28
	* AXI , NJ , NM ,	JTINIT29
	* X(100) , XPRN(100) ,	JTINIT30
	* GAM , RG , PR , PRT ,	JTINIT31
	* SC , TREF , MUREF	JTINIT32
C*		JTINIT33
C*		JTINIT34
C*	MESH COMMON AFTER DISAPPEARANCE OF POTENTIAL CORE	JTINIT35
C*		JTINIT36
C*****	CONTROL COMMON	JTINIT37
C*		JTINIT38
	COMMON /CTRL/	JTINIT39
	* NX1A , CMPRS , QJET , TURBJ , CDEF(10) ,	JTINIT40
	* NPU , NPD , DXC , XU , XDD ,	JTINIT41
	* DSTOR(800)	JTINIT42
C*		JTINIT43
C*****	PROFILE COMMON	JTINIT44
C*		JTINIT45
	COMMON /PROF/ PST(200),Y(200), PRF(200,3)	JTINIT46
C*		JTINIT47
C*****	CONSTANT AND ERROR COMMON	JTINIT48
C*		JTINIT49
	COMMON /CNERR/ BITS , ERR , GC , GCJ , FOOT	JTINIT50

C*		JTINIT51
C*****	BOUNDARY CONDITION COMMON	JTINIT52
C*		JTINIT53
	COMMON /BC/ UEDGE , EEDGE , THEDGE	JTINIT54
C*		JTINIT55
C*****	POTENTIAL CORE COMMON	JTINIT56
C*		JTINIT57
	COMMON /CORED/ XCORE , CORE , CORSTP	JTINIT58
C*		JTINIT59
C*****	SCALER (UNITS CONVERSION) COMMON	JTINIT60
C*		JTINIT61
	COMMON /SCALER/ SP , SV , SLEN	JTINIT62
C*		JTINIT63
C*****	JET PROPERTIES COMMON	JTINIT64
C*		JTINIT65
	COMMON /JET/	JTINIT66
*	B(100) , UC(100) , TC(100) , TIC(100) ,	JTINIT67
*	PIC(100) , WJ(100) , YJ(100) ,	JTINIT68
*	YSDNIC(100)	JTINIT69
	COMMON /JET1/ FLOWJ,ITD,NX,EJET	JTINIT70
	COMMON /UMFISH/ MCHANG,CK,DY1,VMSH , CXPC,CXTP,NRED	JTINIT71
	COMMON /EDGE/ YJETE , SEEDGE	JTINIT72
C*		JTINIT73
	COMMON /MISC/ PM(10) , PLOT	JTINIT74
	COMMON /INPI/ ENTRY1	JTINIT75
	COMMON /JETWQ/	JTINIT76
*	TWD , DIAD , MJETO , TJETO , VJETO ,	JTINIT77
*	PIJETO , TIJETO , NJD	JTINIT78
	COMMON /RCO/ UO, EO, IHO	JTINIT79
	REAL MJETO,MACHD	JTINIT80
	COMMON /CIRL2/	JTINIT81
*	EDGEI , SEI , MERGE , XMERGE , YMERGE ,	JTINIT82
*	SLOPET , SLOPED , CEPII , CEPTD	JTINIT83
	COMMON /MERGET/ MER, MERSTP , XMRG	JTINIT84
	LOGICAL TWD, MERGE , MER , MERSTP	JTINIT85
C*		JTINIT86
	COMMON /MIXER/ MIX,RD(100),XD(100),CF,YR(100)	JTINIT87
	LOGICAL MIX	JTINIT88
	COMMON /FLOBAL/ MAXIT,SUPB,NIT,PSID,YDD,YDC,	JTINIT89
*	P1,P2,UCL,IDL,UPSTRM,CVG	JTINIT90
	LOGICAL SUPB,CVG,UPSTRM	JTINIT91
	COMMON /ACQVVG/ YCD(100),PD(100),INDC(100) , CHOKE, CHOKED	JTINIT92
	LOGICAL CHOKE, CHOKED	JTINIT93
	COMMON /DEIT/ CLSP(100)	JTINIT94
	COMMON /STA2/ MACH2,TS2,SS2,V2,RHDP,OPDX2	JTINIT95
	REAL MACH2	JTINIT96
	COMMON /RCMIX2/ GRADU,TW,MUW,RHOW,PTF,ITE	JTINIT97
	REAL MUW	JTINIT98
	COMMON /CHDDY/ YCH(100),CLSPCH(100),YCHI , HCI1	JTINIT99
	COMMON /OUTMIX/ NXORIG	JTINIT100

COMMON /INNAME/ NB,TAB(5),ND,TAD(4)	JTINIT01
COMMON /SCALD/ SCLD,ALXIM	JTINIT02
LOGICAL SCLD	JTINIT03
COMMON /CRMDD / CJRMDD	JTINIT04
DIMENSION TAB(5),TAD(4)	JTINIT05
DIMENSION TYP(10)	JTINIT06
EQUIVALENCE (IMP(1),PM(1))	JTINIT07
DIMENSION HH(3), CPA(3), CPCD2(3), CPH2D(3)	JTINIT08
DIMENSION CTARN(6)	JTINIT09
DATA CTARN/3HAIR,3HCD2,3HH2D,6H ,6H ,6H /	JTINIT10
DATA HH/41593.74,-164431.94,-119593.66/	JTINIT11
DATA CPA/5.4303,8.929E-4,5.989E-8/	JTINIT12
DATA CPCD2/6.214,5.776E-3,-1.094E-6/	JTINIT13
DATA CPH2D/7.256,1.277E-3,8.735E-8/	JTINIT14
DATA TAB1/1HX,2HXD,2HRD,3HYCB,4HXPRN/	JTINIT15
DATA TAD1/1HY,2HUD,3HTID,3HTHD/	JTINIT16
1 MCHANGE=.TRUE.	JTINIT17
NMSH=71	JTINIT18
CK=1.06064475	JTINIT19
DY1=.001	JTINIT20
CXPC=.02	JTINIT21
CXTP=.02	JTINIT22
NRED=10	JTINIT23
DATA BITS1/03777777777777/	JTINIT24
BITS=BITS1	JTINIT25
2 CALL SETM(1,BITS,MJET,10)	JTINIT26
PF=14.69594	JTINIT27
TF=518.688	JTINIT28
TIJET=0.	JTINIT29
TIE=0.	JTINIT30
TWD=.FALSE.	JTINIT31
CALL SETM(1,BITS,MJETD,4)	JTINIT32
TIJETD=0.	JTINIT33
NJD=50	JTINIT34
MERGE=.FALSE.	JTINIT35
AXI=.TRUE.	JTINIT36
NJ=30	JTINIT37
NM=80	JTINIT38
GAM=1.4	JTINIT39
RG=53.34	JTINIT40
PR=.72	JTINIT41
PRT=1.	JTINIT42
SC=201.6	JTINIT43
TREF=0.	JTINIT44
MUREF=0.	JTINIT45
C* JTINIT46	JTINIT46
3 OJET=.TRUE.	JTINIT47
TURBJ=.TRUE.	JTINIT48
C* JTINIT49	JTINIT49
4 ERR=.FALSE.	JTINIT50

GC=32.174	JTINIT51
GCJ=25036.442	JTINIT52
FONT=12,	JTINIT53
CORE=.FALSE,	JTINIT54
SP=1,	JTINIT55
SV=1,	JTINIT56
SLEN=1.	JTINIT57
C* JTINIT58	
5 CALL SETM(2,RITS,X,100,Y,200)	JTINIT59
CALL SETM(1,0,XPRN,100)	JTINIT60
DPRIN=.FALSE.	JTINIT61
C* JTINIT62	
CALL SETM(1,0.,PM,10)	JTINIT63
IMP(1)=0	JTINIT64
PLNT=0	JTINIT65
ENTRY1=.TRUE.	JTINIT66
MIX=.FALSE.	JTINIT67
CF=.002	JTINIT68
MAXIT=25	JTINIT69
SUPR=.TRUE.	JTINIT70
UPSTRM=.TRUE.	JTINIT71
CVG=.FALSE.	JTINIT72
TOL=1.E-6	JTINIT73
CALL SETM(1,0.,YCB,100)	JTINIT74
CALL SETM(4,RITS,RD,100,XD,100,YR,100,YCD,100)	JTINIT75
CALL SETM(1,0,INDC,100)	JTINIT76
NR=5	JTINIT77
ND = 4	JTINIT78
CALL MOVF(2,TAB1,TAR,5,1,TAD1,TAD,4,1)	JTINIT79
NC=3	JTINIT80
CALL MOVF(1,CTABN,CNAME,6,1)	JTINIT81
DIFF=.FALSE.	JTINIT82
CALL SETM(2,RITS,ALJ,18,ALX,1200)	JTINIT83
CALL SETM(1,7,SCH,6)	JTINIT84
ALF(1)=.99034	JTINIT85
ALE(2)=.00033	JTINIT86
ALE(3)=.00033	JTINIT87
ALJ(1)=.92	JTINIT88
ALJ(2)=.04	JTINIT89
ALJ(3)=.04	JTINIT90
ALJD(1)=.96	JTINIT91
ALJD(2)=.02	JTINIT92
ALJD(3)=.02	JTINIT93
CALL SETM(1,0.,HCPRE,24)	JTINIT94
CALL SETM(1,600.,TCPRE,3)	JTINIT95
CALL MOVF(4,HH,HCPRE,3,1,CPC,CPC(1,1),3,1,CPC(2,CPC(1,2),3,1,	JTINIT96
* CPH2),CPC(1,3),3,1)	JTINIT97
RESTRT=BITS	JTINIT98
NRFS=1	JTINIT99
MIXPRE=.FALSE.	JTINIT100

SCLD = FALSE.
ALXLT= 0001
CJRMND= 1.
RETURN
END

JTINIT01
JTINIT02
JTINIT03
JTINIT04
JTINIT05

```

CJTOUT1      JET/OUTPUT AT STATION 1
SUBROUTINE JTOUT1
C*****
C***** SPECIAL VERSION FOR 12 SPECIES *****
C*****
LOGICAL EOF , ERR
LOGICAL AXI , XPRN , CMPS , QJET , TURBJ , CORE
REAL MJET , ME , MURPF
COMMON /JETWQ/
* TWO , DIAQ , MJETO , TJETO , VJETO ,
* PTJETO , TJETO , NJD
REAL MJETO , MACHQ
COMMON /DIFEQ1/
* NC , CNAME(12) , ALJ(12) , ALJD(12) , ALE(12) , SCM(12) ,
* TCPRF(12) , MCPRF(12) , CPC(3,12)
COMMON /DICTRL/ DIFF , CND(10)
LOGICAL DIFF
COMMON /MOLES / ALX(100,12)
COMMON /RCO/ UO , EO , THO
COMMON /CTRL2/
* EDGFI , SFI , MERGE , XMERGE , YMERGE ,
* SLOPE1 , SLOPED , CEPT1 , CEPTO
COMMON /MERGET/ MER , MERSTP , XMRG
LOGICAL TWO , MERGE , MER , MERSTP
C*
C***** INPUT COMMON
C*
COMMON /INPJET/
* DIAJ , MJET , TJET , PTJET , VJET ,
* TJET ,
* PE , VE , ME , TIE , TE ,
* AXI , NJ , NM ,
* X(100) , XPRN(100) ,
* GAM , RG , PR , PRT ,
* SC , TREP , MUREP
C*
C***** CONTROL COMMON
C*
COMMON /CTRL/
* NXIA , CMPS , QJET , TURBJ , CDEF(10) ,
* NPU , NPD , DXC , XU , XDD ,
* DSTOR(500)
C*
C***** PROFILE COMMON
C*
COMMON /PROF/ PSI(200) , Y(200) , UD(200) , THD(200) , ED(200)
C*
C***** CONSTANT AND ERROR COMMON
C*
COMMON /CNERR/ BITS , ERR , GC , GCJ , FOOT

```

```

JTOUT101
JTOUT102
JTOUT103
JTOUT104
JTOUT105
JTOUT106
JTOUT107
JTOUT108
JTOUT109
JTOUT110
JTOUT111
JTOUT112
JTOUT113
JTOUT114
JTOUT115
JTOUT116
JTOUT117
JTOUT118
JTOUT119
JTOUT120
JTOUT121
JTOUT122
JTOUT123
JTOUT124
JTOUT125
JTOUT126
JTOUT127
JTOUT128
JTOUT129
JTOUT130
JTOUT131
JTOUT132
JTOUT133
JTOUT134
JTOUT135
JTOUT136
JTOUT137
JTOUT138
JTOUT139
JTOUT140
JTOUT141
JTOUT142
JTOUT143
JTOUT144
JTOUT145
JTOUT146
JTOUT147
JTOUT148
JTOUT149
JTOUT150

```

C*		JTOUT151
C*****	BOUNDARY CONDITION COMMON	JTOUT152
C*		JTOUT153
	COMMON /BC/ UEDGE , FEDGE , THEDGE	JTOUT154
C*		JTOUT155
C*****	POTENTIAL CORE COMMON	JTOUT156
C*		JTOUT157
	COMMON /CORED/ XDCORE , CORE , CORSTP	JTOUT158
C*		JTOUT159
C*****	SCALER (UNITS CONVERSION) COMMON	JTOUT160
C*		JTOUT161
	COMMON /SCALER/ SP , SV , SLEN	JTOUT162
C*		JTOUT163
C*****	JET PROPERTIES COMMON	JTOUT164
C*		JTOUT165
	COMMON /JET1/ FLOWJ, YTO, NX, PJET	JTOUT166
	COMMON /JAG/	JTOUT167
	* NAME(10), TITLE(10), IDENT(10), ADDRES(10), IDENT1(10)	JTOUT168
C*		JTOUT169
	COMMON /MIXER/ MIX, RD(100), XD(100), CF, YR(100)	JTOUT170
	LOGICAL MIX	JTOUT171
	COMMON /FLOBA1/ MAXIT, SUPR, NIT, PSID, YDD, YDC,	JTOUT172
	* P1, P2, UCL, TOL, UPSTRM, CVG	JTOUT173
	LOGICAL SUPR, CVG, UPSTRM	JTOUT174
	COMMON /ACONVG/ YCD(100), PD(100), INDC(100), CHOKE, CHOKFD	JTOUT175
	LOGICAL CHOKE, CHOKED	JTOUT176
	COMMON /DEFT/ CLSP(100)	JTOUT177
	COMMON /STA2/ MACH2, TS2, SS2, V2, RHO2, RPDY2	JTOUT178
	REAL MACH2	JTOUT179
	COMMON /RMIX2/ GRADU, TW, MUW, RHDW, PTP, TTE	JTOUT180
	REAL MUW	JTOUT181
C*		JTOUT182
	DIMENSION HEAD1(2,2), HEAD2(2,2), HEAD3(2,2),	JTOUT183
	* FORM1(2), FORM2(2), FORM3(2), TI(200)	JTOUT184
	DIMENSION HEAD4(2,3), FORM4(2)	JTOUT185
	DIMENSION HEAD5(2,2), FORM5(2)	JTOUT186
	EQUIVALENC (TI(1), DSTOR(1))	JTOUT187
	EQUIVALENC (KAXI, AXI), (KQ, QJET), (KMPRS, CMPRS)	JTOUT188
	EQUIVALENC (KTOW, TWO)	JTOUT189
	EQUIVALENC (IMIX, MIX)	JTOUT190
	DATA	JTOUT191
	* HEAD1(1,1)/24HPLANE AXISYMMETRIC/	JTOUT192
	* HEAD2(1,1)/24HISOTHERMAL N-ISOTHERMAL/	JTOUT193
	* HEAD3(1,1)/24HINCOMPRESS. COMPRESSIBLE/	JTOUT194
	DATA	JTOUT195
	* HEAD4(1,1)/36HSINGLE CO-PLANAR CO-ANNULAR /	JTOUT196
	DATA	JTOUT197
	* HEAD5(1,1)/24HFREE JETCONFINED. JPT/	JTOUT198
	DATA BLANK/6H /	JTOUT199
C*		JTOUT100

C* SET VARIABLE HEADINGS	JTOUT101
C*	JTOUT102
1 NAXI=NAXI+1	JTOUT103
NO=KQ+1	JTOUT104
NCMP=KMPRS+1	JTOUT105
NJET=NAXI+KTOW	JTOUT106
IF(NJET.EQ. 2 .AND. AXI) NJETT=1	JTOUT107
KMIX=IMIX+1	JTOUT108
C*	JTOUT109
C* CONSTRUCT HOLLERITH HEADINGS	JTOUT110
C*	JTOUT111
DO 2 L=1,2	JTOUT112
FORM1(L)=HFAD1(L,NAXI)	JTOUT113
FORM2(L)=HEAD2(L,NO)	JTOUT114
FORM4(L)=HFAD4(L,NJET)	JTOUT115
FORM5(L)=HEAD5(L,KMIX)	JTOUT116
2 FORM3(L)=HFAD3(L,NCMP)	JTOUT117
C*	JTOUT118
C*	JTOUT119
XTREF=TREF	JTOUT120
XMUREF=MURFP	JTOUT121
XSC=SC	JTOUT122
IF(TREF.NE. 0.) GO TO 3	JTOUT123
XTREF=BITS	JTOUT124
XMUREF=BITS	JTOUT125
XSC=BITS	JTOUT126
C*	JTOUT127
C* WRITE FIRST SECTION OF OUTPUT	JTOUT128
C*	JTOUT129
3 WRITE(6,100)FORM5,FORM1,FORM2,FORM3,FORM4,	JTOUT130
* NAME,ADDRESS,TITLE,IDENT,IDENT1,	JTOUT131
* TE,DIAJ,GAM,PE,MJET,RC,	JTOUT132
* VE,TJET,PR,ME,PTJET,PRT,TIE,VJET,XSC,TIJET,	JTOUT133
* XTREF,FLOWJ,XMUREF	JTOUT134
IF(TWO) WRITE(6,105) DIAO,MJETO,TJETO,PTJETO,VJETO,TIJETO	JTOUT135
IF(.NOT. TWO) WRITE(6,110)	JTOUT136
C*	JTOUT137
C* CONVERT ED TO TURBULENCE INTENSITY	JTOUT138
C*	JTOUT139
4 CONV=SQRT(2.*GCJ*EJET/3.)/VJET	JTOUT140
DO 5 L=1,NPU	JTOUT141
5 TI(L)=CONV*SQRT(ED(L))	JTOUT142
C* CHECK FOR DIFFUSION CASE	JTOUT143
C*	JTOUT144
IF(DIFF) GO TO 25	JTOUT145
C*	JTOUT146
C* WRITE 2ND SECTION OF OUTPUT	JTOUT147
C*	JTOUT148
10 WRITE(6,150)	JTOUT149
C*	JTOUT150

C*	20 WRITE (6,200) (L,Y(L),PSI(L),UD(L),THD(L),TI(L),L=1,NPU)	JTOUT151
	GO TO 30	JTOUT152
C*		JTOUT153
	25 WRITE (6,151) (CNAME(L),L=1,6)	JTOUT154
	WRITE (6,201) (L,Y(L),PSI(L),UD(L),THD(L),TI(L),	JTOUT155
	* (ALX(L,1),LL=1,6),L=1,NPU),	JTOUT156
	IF(NC.LE,6) GO TO 30	JTOUT157
C*		JTOUT158
C*	SPECIAL OUTPUT FOR NC.GT. 6	JTOUT159
C*		JTOUT160
	NC1 = NC-1	JTOUT161
	DO 29 K=NC1,12	JTOUT162
	29 CNAME(K)= BLANK	JTOUT163
	NUM = (12-NC)*100	JTOUT164
	CALL SFTM(1,RITS,ALX(1,NC1),NUM)	JTOUT165
	WRITE (6,152) (CNAME(L),L=7,12)	JTOUT166
	WRITE (6,202) (L,Y(L),PSI(L),(ALX(L,1),LL=7,12),L=1,NPU)	JTOUT167
C*		JTOUT168
C*	IF CONFINED MIXING,PRINT PSI OF DUCT SURFACE	JTOUT169
C*		JTOUT170
	30 IF(MIX) WRITE (6,250) YR(1),PSIO	JTOUT171
C*		JTOUT172
C*	***** FORMAT STATEMENTS *****	JTOUT173
C*		JTOUT174
	100 FORMAT(1H1,23X,3H* ,2A6,11H PROGRAM *///,	JTOUT175
	* 5H * ,2A6,11H * * ,2A6,10H * * ,2A6,	JTOUT176
	* 10H * JET///,	JTOUT177
	* 16X,6H* ,2A6,2X,20HMIXING REGION *///,	JTOUT178
	* 10A6/,10A6/,10A6/,10A6/,10A6///,	JTOUT179
	* 17X,37H* INPUT AND INITIAL CONDITIONS *///,	JTOUT180
	* 2X,19HEXTERNAL CONDITIONS,3X,24HJET DISCHARGE PARAMETERS,	JTOUT181
	* 5X,14HGAS PROPERTIES//,2X,4HTE =,F14.3,6X,	JTOUT182
	* 6HDIAJ =,F13.5,5X,6HGM =,F13.5/,2X,	JTOUT183
	* 4HPE =,1PF14.4,6X,6HJET =,0PF13.4,5X,	JTOUT184
	* 6HRG =,F13.5/,2X,4HVE =,F14.3,6X,6HJET =,F13.3,	JTOUT185
	* 5X,6HPR =,F13.5/,2X,4HMF =,F14.4,6X,6HJET =,	JTOUT186
	* 1PE13.4,5X,6HPR =,0PF13.5/,2X,4HTIF =,1PE14.4,	JTOUT187
	* 6X,6HVJET =,0PF13.3,5X,6HSC =,F13.3/,26X,	JTOUT188
	* 6HTIJET =,1PE13.4,5X,6HREF =,0PF13.3/,26X,6HFLOWJ =,1PE13.4,5X,	JTOUT189
	* 6HMURFF =,F13.4/)	JTOUT190
	110 FORMAT(///)	JTOUT191
	105 FORMAT(26X,7HDIAJ =,F12.5/26X,7HJET =,F12.4/	JTOUT192
	* 26X,7HTJET =,F12.3/26X,7HPTJET =,1PF12.4/	JTOUT193
	* 26X,7HVJET =,0PF12.3/26X,7HTIJET =,1PE12.4/)	JTOUT194
C*		JTOUT195
	150 FORMAT(23X,26H* INITIAL PROFILES *//	JTOUT196
	* 1X,1HN,7X,1HY,11X,3HPSI,13X,1HU,11X,5HNETA,10X,2HTI//)	JTOUT197
C*		JTOUT198
	151 FORMAT(53X,26H* INITIAL PROFILES *//	JTOUT199
		JTOUT100

*IX,1HN,7X,1HY,11X,3HPSI,10X,1HU,8X,5HTheta,9X,2HTI:	JTOUT101
* 7X,A6,5X,A6,5X,A6,5X,A6,5X,A6,5X,A6//)	JTOUT102
152 FORMAT(1H1,32X,26H* INITIAL . PROFILES *//	JTOUT103
* IX,1HN,7X,1HY,11X,3HPSI,7X,A6,5(5X,A6)//)	JTOUT104
200 FORMAT(I3,1PE12.4,E13.4,OPF14.8,F14.8,1PE16.7)	JTOUT105
201 FORMAT(I3,1PE12.4,E13.4,OPF11.8,F11.8,1PE15.7,6E11.4)	JTOUT106
202 FORMAT(I3,1PE12.4,E13.4,6E11.4)	JTOUT107
250 FORMAT(/4HALL,1PE11.4,E13.4)	JTOUT108
C*	JTOUT109
1000 RETURN	JTOUT110
END	JTOUT111

CJTOUTS	SUMMARY PRINT OF JET PROPERTIES	JTOUTS01
	SUBROUTINE JTOUTS	JTOUTS02
C*****		JTOUTS03
C*****	SPECIAL VERSION FOR 12 SPECIES	JTOUTS04
C*****		JTOUTS05
	LOGICAL ENF , ERR	JTOUTS06
	LOGICAL TAPIN,TAPOT	JTOUTS07
	LOGICAL AXI , XPRN , CMPS , QJET , TURBJ , CORE	JTOUTS08
	LOGICAL ENDJOB,ENDJW	JTOUTS09
	INTEGER XPRN1(100)	JTOUTS10
	EQUIVALENCE (XPRN,XPRN1)	JTOUTS11
	INTEGER PLNT	JTOUTS12
	REAL MJET , ME , MUREF	JTOUTS13
	COMMON /ADAM02/ ENDJOB , DUM02(3)	JTOUTS14
	COMMON /DIFEQ1/	JTOUTS15
	* NC , CNAME(12) , ALJ(12) , ALJ2(12) , ALE(12) , SCM(12) ,	JTOUTS16
	* TCPRF(12) , MCPRF(12) , CPC(3,12)	JTOUTS17
	COMMON /DICTRL/ DIFF , CND(10)	JTOUTS18
	LOGICAL DIFF	JTOUTS19
	COMMON /MOIES / ALX(100,12)	JTOUTS20
	COMMON /RCM2L / ALEDGE(12),ALO(12)	JTOUTS21
	COMMON /JETTWO/	JTOUTS22
	* TWO , DIAO , MJETO , TJETO , VJETO ,	JTOUTS23
	* PTJETO , TIJETO , NJO	JTOUTS24
	REAL MJETO,MACHO	JTOUTS25
	COMMON /BCO/ UO, EO, THO	JTOUTS26
	COMMON /CTRL2/	JTOUTS27
	* EDGFI , SFI , MERGE , XMERGE , YMERGE ,	JTOUTS28
	* SLOPEI , SLOPED , CEPTI , CEPIO	JTOUTS29
	COMMON /MERGET/ MER, MERSTP , XMRG	JTOUTS30
	LOGICAL TWO, MERGE , MER , MERSTP	JTOUTS31
	COMMON /TOFILE/ TAPIN,TAPOT	JTOUTS32
	COMMON /PARAM/	JTOUTS33
	* U(200),T(200),TOT(200),XMACH(200),PTOT(200),TTD(200),	JTOUTS34
	* PTD(200),UDC(200),DUMP9(209)	JTOUTS35
C*		JTOUTS36
C*****	INPUT COMMON	JTOUTS37
C*		JTOUTS38
	COMMON /INPJET/	JTOUTS39
	* DIAJ , MJET , TJET , PTJET , VJET ,	JTOUTS40
	* TIJET ,	JTOUTS41
	* PE , VE , ME , TIF , TE ,	JTOUTS42
	* AXI , NJ , NM ,	JTOUTS43
	* X(100) , XPRN(100) ,	JTOUTS44
	* GAM , RG , PR , PRI ,	JTOUTS45
	* SC , TREF , MUREF	JTOUTS46
C*		JTOUTS47
C*****	CONTROL COMMON	JTOUTS48
C*		JTOUTS49
	COMMON /CTRL/	JTOUTS50

* NXTA	, CMPRS	, QJET	, TURBJ	, COEF(10)	JTOUTS51
* NPU	, NPD	, DXC	, XU	, XDD	JTOUTS52
* DSTOR(800)					JTOUTS53
C*					JTOUTS54
C*****	PROFILE COMMON				JTOUTS55
C*					JTOUTS56
	COMMON /PRDF/	PSI(200)	Y(200)	UD(200)	JTOUTS57
					JTOUTS58
C*					JTOUTS59
C*****	CONSTANT AND ERROR COMMON				JTOUTS60
C*					JTOUTS61
	COMMON /CNERR/	BITS	, ERR	, GC	JTOUTS62
C*					JTOUTS63
C*****	BOUNDARY CONDITION COMMON				JTOUTS64
C*					JTOUTS65
	COMMON /BC/	UEDGE	, EEDGE	, THEDGE	JTOUTS66
C*					JTOUTS67
C*****	POTENTIAL CORE COMMON				JTOUTS68
C*					JTOUTS69
	COMMON /CORED/	XCORE	, CORE	, CORSTP	JTOUTS70
C*					JTOUTS71
C*****	SCALER (UNITS CONVERSION) COMMON				JTOUTS72
C*					JTOUTS73
	COMMON /SCALER/	SP	, SV	, SLEN	JTOUTS74
C*					JTOUTS75
C*****	JET PROPERTIES COMMON				JTOUTS76
C*					JTOUTS77
	COMMON /JET/				JTOUTS78
	* B(100)	, UC(100)	, TC(100)	, TIC(100)	JTOUTS79
	* PTC(100)	, WJ(100)	, YJ(100)		JTOUTS80
	* YSONIC(100)				JTOUTS81
	COMMON /JET1/	FLWJ	, T10	, NX	JTOUTS82
	COMMON /JET2/	TTC(100)			JTOUTS83
	COMMON /TAG/				JTOUTS84
	* NAME(10)	, TITLE(10)	, IDENT(10)	, ADDRES(10)	JTOUTS85
	COMMON /MISC/	PM(10)	, PLNT		JTOUTS86
C*					JTOUTS87
	COMMON /MIXER/	MIX	, RD(100)	, XD(100)	JTOUTS88
	LOGICAL MIX				JTOUTS89
	COMMON /FLQBAL/	MAXIT	, SUPB	, NIT	JTOUTS90
					JTOUTS91
					JTOUTS92
	COMMON /ACNVVG/	YCD(100)	, PD(100)	, INDC(100)	JTOUTS93
	LOGICAL CHDKE	, CHDKE			JTOUTS94
	COMMON /DETT/	CLSP(100)			JTOUTS95
	COMMON /STA2/	MACH2	, TS2	, SS2	JTOUTS96
	REAL MACH2				JTOUTS97
	COMMON /BGMIX2/	GRADU	, IW	, MUW	JTOUTS98
	REAL MUW				JTOUTS99
	COMMON /THERM/	GMC(200)	, CP(200)		JTOUTS00
	COMMON /MIXPRP/	MA2(100)	, VF2(100)	, TE2(100)	

REAL MA2	JTOUTS01
COMMON /THRST/ WV(100)	JTOUTS02
COMMON /CBODY/ YCB(100),CLSPCB(100),YCB1 , UCL1	JTOUTS03
COMMON /OUTMIX/ NXORIG	JTOUTS04
COMMON /JET3/ STADD, WV, STATF	JTOUTS05
LOGICAL STADD,STATF	JTOUTS06
C* DIMENSION TI(200)	JTOUTS07
DIMENSION LF1(3)	JTOUTS08
DIMENSION HEAD1(2,2), FORM1(2)	JTOUTS09
EQUIVALENCE (TI(1),DSTOR(1))	JTOUTS10
EQUIVALENCE (IMIX, MIX)	JTOUTS11
DATA LF1(1)/18HSUBSONSUPSONCHOKED/	JTOUTS12
DATA HEAD1(1,1)/24HFREE JET CONFINED JET/	JTOUTS13
DATA ENDJW/F/	JTOUTS14
C* GO TO 1	JTOUTS15
ENTRY JTPWR	JTOUTS16
NXTA=NX-1	JTOUTS17
PLOT=0	JTOUTS18
ENDJW=.TRUE.	JTOUTS19
1 IF(PLOT.GT. 0 .OR. TAPOT) CALL JTFILE(2,DUM)	JTOUTS20
KMIX= IMIX+1	JTOUTS21
DO 6 L=1,2	JTOUTS22
6 FORM1(L)=HEAD1(L,KMIX)	JTOUTS23
DO 7 L=1,NXTA	JTOUTS24
IF(INDC(L).EQ.0) INDIC(L)=LF1(1)	JTOUTS25
IF(INDC(L).EQ.1) INDIC(L)=LF1(2)	JTOUTS26
IF(INDC(L).EQ.2) INDIC(L)=LF1(3)	JTOUTS27
7 CONTINUE	JTOUTS28
TERMD=.5*DIAJ	JTOUTS29
CALL FMPYC(1,TERMD,YCB,YCB,NXORIG)	JTOUTS30
WRITE (6,100)	JTOUTS31
NSTART=1	JTOUTS32
NLINES=NXTA	JTOUTS33
IF(NLINES.GT.50) NLINES=50	JTOUTS34
NL=NLINES	JTOUTS35
2 WRITE (6,110) FORM1,IDENT,IDENT1	JTOUTS36
3 WRITE (6,120)	JTOUTS37
NEND=NL	JTOUTS38
ASSIGN 10 TO LGO	JTOUTS39
5 WRITE (6,200) (L,X(L),B(L),YJ(L),UC(L),TC(L),	JTOUTS40
* TIC(L),PTC(L),ITC(L),YSONIC(L),WJ(L),L=NSTART,NEND)	JTOUTS41
GO TO LGO , (10,20)	JTOUTS42
C* JTOUTS43	JTOUTS43
C*** CHECK FOR ADDITIONAL LINES	JTOUTS44
C* JTOUTS45	JTOUTS45
10 IF(NXTA.LE.50) GO TO 20	JTOUTS46
NSTART=NEND+1	JTOUTS47
NEND=NXTA	JTOUTS48
	JTOUTS49
	JTOUTS50

AD-A045 627

GENERAL ELECTRIC CO CINCINNATI OHIO AIRCRAFT ENGINE GROUP F/G 21/5
DEVELOPMENT OF EMISSIONS MEASUREMENT TECHNIQUES FOR AFTERBURNIN--ETC(U)
OCT 75 W C COLLEY, D R FERGUSON, M A SMITH F33615-73-C-2047

UNCLASSIFIED

R75AEG459

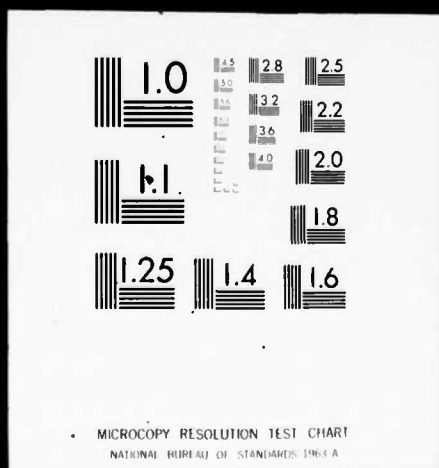
AFAPL-TR-75-52-SUPPL-2

NL

4 of 5
AD-A045 627



4 OF 5
ADA045 627



WRITE (6,100)	JTOUTS51
WRITE (6,120)	JTOUTS52
ASSIGN 20 TO LGO	JTOUTS53
C* GO TO 5	JTOUTS54
C* IF CONFINED JET CASE, PRINT CONFINED JET OUTPUT	JTOUTS55
C* 20 IF(.NOT. MIX) GO TO 21	JTOUTS56
WRITE (6,100)	JTOUTS57
NSTART=1	JTOUTS58
NLINES=NV	JTOUTS59
IF(NLINES.GT.50) NLINES=50	JTOUTS60
NL=NLINES	JTOUTS61
22 WRITE (6,110) FORM1,IDENT,IDENT1	JTOUTS62
23 WRITE (6,140)	JTOUTS63
NEND=NL	JTOUTS64
ASSIGN 30 TO LGO	JTOUTS65
25 WRITE(6,2000) (L,XD(L),RD(L),YCB(L),YR(L),YCD(L),	JTOUTS66
* PD(L),INDC(L),WV(L),MA2(L),VE2(L),TF2(L),L=NSTART,NEND)	JTOUTS67
GO TO LGO , (30,21)	JTOUTS68
C* 30 IF(NV.LE.50) GO TO 21	JTOUTS69
NSTART=NEND+1	JTOUTS70
NEND=NV	JTOUTS71
WRITE (6,100)	JTOUTS72
WRITE (6,140)	JTOUTS73
ASSIGN 21 TO LGO	JTOUTS74
GO TO 25	JTOUTS75
C* 21 IF(.NOT. ENDJW) RETURN	JTOUTS76
ENDJOB=.TRUE.	JTOUTS77
CALL ERROR	JTOUTS78
C* ***** FORMAT STATEMENTS *****	JTOUTS79
C* 100 FORMAT (1H1)	JTOUTS80
110 FORMAT(39X,49H* JET ANALYSIS PROGRAM *//	JTOUTS81
* 48X,2A6,6X,6H MIXING//	JTOUTS82
* 30X,10A6/,30X,10A6//)	JTOUTS83
120 FORMAT(39X,51H* SUMMARY = STATION DATA = JET PROPERTIES *	JTOUTS84
*//,4X,14N,10X,1HX,13X,1HB,11X,2HYJ,12X,2HUC,	JTOUTS85
* 12X,2HTC,11X,3HTIC,7X,3HPTC,8X,3HTC,5X,6HYSOHC,6X,2HWJ//)	JTOUTS86
200 FORMAT(3X,13,F14.5,F13.5,F14.5,F15.7,F14.7,1PF15.6,	JTOUTS87
* 0PF9.6,3F10.6)	JTOUTS88
140 FORMAT(4X,14N,6X,2HXD,8X,2HRD,7X,3HYFB,8X,2HYD,9X,3HYCD,	JTOUTS89
* 9X,2HPD,5X,4HFLDN,2X,6HTHRUST,6X,3HMA2,5X,3HVF2,	JTOUTS90
* 6X,3HTE2//)	JTOUTS91
2000 FORMAT(3X,13,3F10.4,2F12.7,F9.4,1X,A6,F11.3,F7.4,F9.3,F8.2)	JTOUTS92
C* JTOUTS93	JTOUTS93
	JTOUTS94
	JTOUTS95
	JTOUTS96
	JTOUTS97
	JTOUTS98
	JTOUTS99
	JTOUTS00

C*		JTOUTS01
C*	***** JTOUTP ENTRY *****	JTOUTS02
C*		JTOUTS03
C*	PROFILE PRINTOUT	JTOUTS04
C*		JTOUTS05
	ENTRY JTOUTP	JTOUTS06
	PK=PF	JTOUTS07
	IF(MIX) PK=P2	JTOUTS08
C*		JTOUTS09
C*		JTOUTS10
	50 NSTA=NX	JTOUTS11
	51 XSTA=X(NX)	JTOUTS12
C*		JTOUTS13
C*	CONVERT TURBULENCE ENERGIES TO INTENSITIES	JTOUTS14
C*		JTOUTS15
	52 CON1=SQRT(2.*GCJ*EJET/3.)/VJET	JTOUTS16
	DO 53 L=1,NPD	JTOUTS17
	53 TI(L)=CON1*SQRT(ED(L))	JTOUTS18
C*		JTOUTS19
C*	COMPUTE ADDITIONAL DIMENSIONLESS AND DIMENSIONAL PROFILES	JTOUTS20
C*		JTOUTS21
	UC1=1./UC(NX)	JTOUTS22
	TMPT=1./(TTO-TE)	JTOUTS23
	PTPT=1./(PTJET-PE)	JTOUTS24
	CALL FMPYC(1,TJET,TMD,T,NPD)	JTOUTS25
	CALL GAMCP(T,GMC,CP,RG,1,NPD)	JTOUTS26
	DO 600 L=1,NPD	JTOUTS27
	U(L)=VJET*UD(L)	JTOUTS28
	IF(.NOT. CMPRS) GO TO 600	JTOUTS29
	GAM=GMC(L)	JTOUTS30
	GM2=2./(GAM-1.)	JTOUTS31
	GMM=GAM/(GAM-1.)	JTOUTS32
	CPJ=CP(L)	JTOUTS33
	GCJCP=1./(GCJ*CPJ)	JTOUTS34
	TOT(L)=T(L)+.5*U(L)*U(L)*GCJCP+EJET*ED(L)/CPJ	JTOUTS35
	XMACH(L)=U(L)/SQRT(GAM*RG*GC*T(L))	JTOUTS36
	PTOT(L)=PK*(1.+XMACH(L)**2/GM2)**GMM	JTOUTS37
	TTD(L)=(TOT(L)-TE)*TMPT	JTOUTS38
	PTD(L)=(PTOT(L)-PE)*PTPT	JTOUTS39
	600 CONTINUE	JTOUTS40
	IF(.NOT. CMPRS) CALL SEIM(5,BITS,TOT,NPD,XMACH,NPD,	JTOUTS41
	* PTOT,NPD,TTD,NPD,PTD,NPD)	JTOUTS42
C*		JTOUTS43
	IF(PLDT.GT.0 .OR. (XPRN(NX) .AND. TAPDY)) CALL JTFILE(1,XSTA)	JTOUTS44
	IF(XPRN1(NX) .LT. 0) GO TO 111	JTOUTS45
	55 WRITE (6,300)	JTOUTS46
	NSTART=1	JTOUTS47
	NLINES=NPD	JTOUTS48
	IF(NLINES.GT.50) NLINES=50	JTOUTS49
	NL=NLINES	JTOUTS50

56	WRITE (6,310) NSTA,XSTA,PK	JTOUTS51
	NEND=NL	JTOUTS52
	ASSIGN 60 TO LGO1	JTOUTS53
57	WRITE (6,320) (L,Y(L),PSI(L),UD(L),THD(L),TI(L),ITD(L),PTD(L),	JTOUTS54
	* XMACH(L),II(L),T(L),TOT(L),PTOT(L),L=NSTART,NEND)	JTOUTS55
	GO TO LGO1 , (60,90)	JTOUTS56
C*		JTOUTS57
C*	CHECK FOR ADDITIONAL PRINT LINES	JTOUTS58
C*		JTOUTS59
60	IF(NPD.LE. 50) GO TO 90	JTOUTS60
	WRITE (6,100)	JTOUTS61
	WRITE (6,310) NSTA,XSTA,PK	JTOUTS62
	NRMN=NPD-NI	JTOUTS63
	IF(NRMN.GT.50) GO TO 65	JTOUTS64
	NSTART=NEND+1	JTOUTS65
	NEXT=MINO(50,NRMN)	JTOUTS66
	NI=NI+NEXT	JTOUTS67
	ASSIGN 90 TO LGO1	JTOUTS68
	NEND=NL	JTOUTS69
	GO TO 57	JTOUTS70
65	NSTART=NEND+1	JTOUTS71
	NEND=NL+50	JTOUTS72
	NI=NI+50	JTOUTS73
	GO TO 57	JTOUTS74
C*		JTOUTS75
C*		JTOUTS76
C*	PRINT CONCENTRATION PROFILES	JTOUTS77
C*		JTOUTS78
90	IF(.NOT. DIFF) GO TO 1111	JTOUTS79
	WRITE (6,300)	JTOUTS80
	NSTART=1	JTOUTS81
	NLINES=NPD	JTOUTS82
	IF(NLINES.GT.50) NLINES=50	JTOUTS83
	NI=NLINES	JTOUTS84
1112	WRITE (6,311) NSTA,XSTA,PK,CNAME	JTOUTS85
	NEND=NL	JTOUTS86
	ASSIGN 1120 TO LGO1	JTOUTS87
1113	WRITE (6,321) (L,Y(L),(ALX(L,LL),LL=1,12),I=NSTART,NEND)	JTOUTS88
	GO TO LGO1 , (1120,1111)	JTOUTS89
C*		JTOUTS90
1120	IF(NPD.LE.50) GO TO 1111	JTOUTS91
	WRITE (6,300)	JTOUTS92
	WRITE (6,311) NSTA,XSTA,PK,CNAME	JTOUTS93
	NRMN=NPD-NI	JTOUTS94
	IF(NRMN.GT.50) GO TO 1114	JTOUTS95
	NSTART=NEND+1	JTOUTS96
	NEXT=MINO(50,NRMN)	JTOUTS97
	NI=NI+NEXT	JTOUTS98
	ASSIGN 1111 TO LGO1	JTOUTS99
	NEND=NL	JTOUTS00

GO TO 1113	JTOUTS01
1114 NSTART=NL+50	JTOUTS02
NEND=NL+50	JTOUTS03
NL=NL+50	JTOUTS04
GO TO 1113	JTOUTS05
C***** FORMAT STATEMENTS *****	JTOUTS06
C*	JTOUTS07
300 FORMAT(1H1,50X,28H* JET ANALYSIS PROGRAM **//)	JTOUTS08
310 FORMAT(26X,16HPROFILES== STA (,I3,6H) X=,F10.5,	JTOUTS09
* 3X,9HPRESSURE=,F10.4//	JTOUTS10
* 22X,1H*,21X,21H** DIMENSIONLESS **,22X,	JTOUTS11
* 2H**,12X,19H** DIMENSIONAL **,9X,1H*/	JTOUTS12
* 2X,1HN,6X,1HY,9X,3HPSI,8X,2HJD,7X,3HTHD,9X,2HTI,8X,3HTTD,8X,	JTOUTS13
* 3HPTD,6X,4HMACH,8X,1HU,11X,1HT,9X,3HTDT,7X,4HPTDT//)	JTOUTS14
311 FORMAT(12X,22HMOLE FRACTIONS== STA (,I3,6H) X=,	JTOUTS15
* F10.5,3X,9HPRESSURE=,F10.4//	JTOUTS16
* 2X,1HN,6X,1HY,5X,12(A6,3X)//)	JTOUTS17
320 FORMAT(1X,I3,F11.5,1PE11.4,0PF9.6,F10.6,1PF12.5,0PF10.7,	JTOUTS18
* F11.7,F9.5,3F11.4,F10.4)	JTOUTS19
321 FORMAT(1X,I3,F9.5,12F9.6)	JTOUTS20
C*	JTOUTS21
1111 RETURN	JTOUTS22
END	JTOUTS23

CJTSTP	JFT--SOLUTION ROUTINE	JTSTP001
	SUBROUTINE JTSTEP	JTSTP002
C*****		JTSTP003
C*****	SPECIAL VERSION FOR 12 SPECIFS	JTSTP004
C*****		JTSTP005
	LOGICAL ICYCLE	JTSTP006
	INTEGER TWOJ, ITWO	JTSTP007
	LOGICAL SUPC, SUPSTP	JTSTP008
	LOGICAL SUBSON	JTSTP009
	LOGICAL TROUBL	JTSTP010
	LOGICAL DPRINT	JTSTP011
	LOGICAL LAST, CORSTP, ADDP, ENTRY1, IER	JTSTP012
	LOGICAL ENF, ERR	JTSTP013
	LOGICAL AXI, XPRN, CMPS, QJET, TURBJ, CORE	JTSTP014
	REAL KCP, MIL, MUEFF, MACH	JTSTP015
	REAL MJET, ME, MUREF	JTSTP016
	COMMON /RSTART/ NREG, RESTRT, NRES, MIXPRE	JTSTP017
	LOGICAL MIXPRE	JTSTP018
	COMMON /MDIUP / ALXU(100,12), DALXU(100,12), DTKE(200)	JTSTP019
	COMMON /DIFEQ1/	JTSTP020
	* NC, CNAME(12), ALJ(12), ALJD(12), ALF(12), SCM(12);	JTSTP021
	* TCFRE(12), HCFRE(12), CPC(3,12)	JTSTP022
	COMMON /DICTRL/ DIFF, CND(10)	JTSTP023
	LOGICAL DIFF	JTSTP024
	COMMON /MDIES / ALX(100,12)	JTSTP025
	COMMON /HCDL / ALEDGE(12), ALO(12)	JTSTP026
	COMMON /JETTWO/	JTSTP027
	* TWO, DIAD, MJETO, TJETO, VJETO	JTSTP028
	* PTJETO, TJETO, NJD	JTSTP029
	REAL MJETO, MACHD	JTSTP030
	COMMON /BCO/ UO, EO, THO	JTSTP031
	COMMON /CTRL2/	JTSTP032
	* EDGE1, SFI, MERGE, XMERGE, YMERGE;	JTSTP033
	* SLOPE1, SLOPE2, CEPT1, CEPT2	JTSTP034
	COMMON /MERGET/ MER, MERSTP, XMRG	JTSTP035
	LOGICAL TWO, MERGE, MER, MERSTP	JTSTP036
	COMMON /SETNEW/ LEDGE, LCOEN	JTSTP037
	COMMON /INP1 / ENTRY1	JTSTP038
	COMMON /MISC/ PM(10)	JTSTP039
	COMMON /PARAM/	JTSTP040
	* AL(200), RF(200), GM(200),	JTSTP041
	* EPS(200), DL(200)	JTSTP042
	* VAR(200), DVAR(200),	JTSTP043
	* SM1(200), NM1, SM(200), NM	JTSTP044
	* DX,	JTSTP045
	* B1, C11, D1	JTSTP046
	* AN, BN, DN	JTSTP047
	COMMON /PARAM1/ ETA(200)	JTSTP048
C*		JTSTP049
C*****	INPUT COMMON	JTSTP050

C*	COMMON /INPJET/	JTSTP051
	* DIAJ , MJET , TJET , PTJET , VJET ,	JTSTP052
	* TIJET	JTSTP053
	* PE , VE , ME , TIE , TE ,	JTSTP054
	* AXI , NJ , NMAX ,	JTSTP055
	* XJ(100) , XPRN(100) ,	JTSTP056
	* GAM , RG , PR , PRT ,	JTSTP057
	* SC , TREF , MUREP	JTSTP058
C*		JTSTP059
C*****	CONTROL COMMON	JTSTP060
C*		JTSTP061
	COMMON /CTRL/	JTSTP062
	* NXTA , CMPS , QJET , TURBJ , COEF(10) ,	JTSTP063
	* NPU , NPD , DXC , XU , XDD ,	JTSTP064
	* DSTOR(800)	JTSTP065
C*		JTSTP066
C*****	PROFILE COMMON	JTSTP067
C*		JTSTP068
	COMMON /PRNF/ PSI(200),Y(200),UD(200),THD(200),ED(200)	JTSTP069
C*		JTSTP070
C*****	CONSTANT AND ERROR COMMON	JTSTP071
C*		JTSTP072
	COMMON /CNFRR/ BITS , FRR , GC , GCJ , FOOT	JTSTP073
C*		JTSTP074
C*****	BOUNDARY CONDITION COMMON	JTSTP075
C*		JTSTP076
	COMMON /BC/ UEDGE , FEDGE , THEDGE	JTSTP077
C*		JTSTP078
C*****	POTENTIAL CORE COMMON	JTSTP079
C*		JTSTP080
	COMMON /CORED/ XCORE , CORE , CORSTP	JTSTP081
C*		JTSTP082
	COMMON /SUPER/ SUPC,SUPSTP,XSUP	JTSTP083
C*****	SCALER (UNITS CONVERSION) COMMON	JTSTP084
C*		JTSTP085
	COMMON /SCALER/ SP , SV , SLEN	JTSTP086
C*		JTSTP087
C*****	JET PROPERTIES COMMON	JTSTP088
C*		JTSTP089
	COMMON /JET/	JTSTP090
	* B(100) , UC(100) , TC(100) , TIE(100) ,	JTSTP091
	* PTC(100) , WJ(100) , YJ(100)	JTSTP092
	COMMON /JET1/ FLOWJ,TIO,NX,EJET	JTSTP093
	COMMON /ERASE/ YD(200),TKE(200),T(200),DUDY(200)	JTSTP094
C*		JTSTP095
	COMMON /PROPJT/	JTSTP096
	* P , PRL , PRTT , RGAS , SCC	JTSTP097
	* TREFF , VSRFF , WACH , XLC ,	JTSTP098
	* REFL , C , CHI , RNORM ,	JTSTP099
		JTSTP100

* RHO(200) , MUL(200) , KCP(200) ,	JTSTP101
* MUEFF(200) , XLN(200) , DK(200) , RETURB(200)	JTSTP102
COMMON /XPRIN/ DPHIN	JTSTP103
COMMON /EDGE/ YJETE , SFEDGE	JTSTP104
COMMON /UMFSH/ DUMUI(4),CXPC,CXTP,NRFD	JTSTP105
C* COMMON /MIXER/ MIX,RDD(100),XD(100),CF,YR(100)	JTSTP106
LOGICAL MIX	JTSTP107
COMMON /FLOPAL/ MAXIT,SUPB,NIT,PSID,YDD,YDC,	JTSTP108
* P1,P2,UCL,TOL,UPSTRM,CVG	JTSTP109
LOGICAL SUPB,CVG,UPSTRM	JTSTP110
COMMON /ACONVG/ YCD(100),PD(100),INDC(100), CHOKE, CHOKED	JTSTP111
LOGICAL CHOKE, CHOKED	JTSTP112
COMMON /DFIT/ CLSP(100)	JTSTP113
COMMON /STA2/ MACH2,TS2,SS2,V2,RHQ2,NPDX2	JTSTP114
REAL MACH2	JTSTP115
COMMON /ACMIX2/ GRADU,TW,MUW,RHOW,PTF,TTE	JTSTP116
REAL MUW	JTSTP117
COMMON /THERM/ GMC(200),CP(200)	JTSTP118
COMMON /CHODY/ YCB(100),CLSPCB(100),YCB1 , UCL1	JTSTP119
COMMON /OUTMIX/ NXORIG	JTSTP120
COMMON /SCALFD/ SCLD,ALXLTM	JTSTP121
LOGICAL SCLD	JTSTP122
C* JTSTP123	
C* JTSTP124	
JTSTP125	
DIMENSION HU(200),EU(200),THU(200)	JTSTP126
DIMENSION UK(200)	JTSTP127
DIMENSION NAM(4)	JTSTP128
DIMENSION YU(200)	JTSTP129
EQUIVALENCE (YU(1),DSTOR(401))	JTSTP130
EQUIVALENCE (UK(1),DSTOR(1))	JTSTP131
EQUIVALENCE	JTSTP132
* (C1,COEF(1)),(C2,COEF(2)),(C3,COEF(3)),(C4,COEF(4)),	JTSTP133
* (C5,COEF(5)),(C6,COEF(6)),(C7,COEF(7))	JTSTP134
EQUIVALENCE (C9,COEF(9))	JTSTP135
EQUIVALENCE (ITWO,IWD)	JTSTP136
DATA NAM/1WY,2HSM,3HXLN,2HUD,2REC,3MTHD/	JTSTP137
* DATA ENTRY1/1/	JTSTP138
C* JTSTP139	
C* TEST FOR 1-ST STEP(ENTRY1=1)	JTSTP140
C* JTSTP141	
ASSIGN 16 TO LGDP	JTSTP142
IF(ENTRY1 .AND. (RESTR1.EQ.BITS)) CORSTP=.FALSE.	JTSTP143
5 IF(ENTRY1) DX=AMINI(CXPC*B(NRES),.5*DXC)	JTSTP144
IF(RFSTRT.EQ. BITS) GO TO 5326	JTSTP145
IF(NREC.GT. 1) DX=AMINI(CXTP*B(NRES),.5*DXC)	JTSTP146
5326 IF(DXC.EQ.0.) GO TO 5327	JTSTP147
IF(.NOT. CORSTP) DX=AMINI(DX, .5*DXC)	JTSTP148
5327 CONTINUE	JTSTP149
IF(.NOT. ENTRY1) GO TO 9	JTSTP150

C*		JTSTP151
C*----	ENTRY 1--INITIALIZE FOR 1-ST STEP:	JTSTP152
C*		JTSTP153
	CALL SETM(1,0,ETA,200)	JTSTP154
	ICYCLE=.FALSE.	JTSTP155
	TWOJ=ITWO+1	JTSTP156
	CALL MOVE(5,PSI,SM,NPU,1,PSI,SM1,NPU,1,UD,UU,NPU,1,TWD,THU,NPD,1,	JTSTP157
	* ED,EU,NPU,1)	JTSTP158
	IF(.NOT. DIFF) GO TO 7000	JTSTP159
	DO 7001 I=1,NC	JTSTP160
	CALL MOVE(1,ALX(1,L),ALXU(1,L),NPU,1)	JTSTP161
7001	CONTINUE	JTSTP162
7000	NH=NPU	JTSTP163
	NH1=NPU	JTSTP164
	EC=ED(1)	JTSTP165
	THC=THD(1)	JTSTP166
	IF(.NOT. THRBJ) CALL SETM(2,EC,ED,200,EU,200)	JTSTP167
	IF(.NOT. QJET) CALL SETM(2,THC,THD,200,THU,200)	JTSTP168
	EPS1=0.	JTSTP169
	IF(AXI) EPS1=1.	JTSTP170
C*		JTSTP171
4	LAST=.FALSE.	JTSTP172
	ADDP=.FALSE.	JTSTP173
	IF(RESTRI.NE. BITS) GO TO 6	JTSTP174
	CORSTP=.FALSE.	JTSTP175
	SUPSTP=.FALSE.	JTSTP176
	MERSTP=.FALSE.	JTSTP177
	NREG=1	JTSTP178
	SUBSON=.TRUE.	JTSTP179
	IF(MJET,GE,1.) SUBSON=.FALSE.	JTSTP180
6	X=XU	JTSTP181
	LCORFN=1	JTSTP182
	LEDGF=0	JTSTP183
	CALL SETM(1,C1,AL,200)	JTSTP184
	CALL MOVE(1,Y,YU,NPU,1)	JTSTP185
C*		JTSTP186
C*	INCREMENT STEP COUNTER, X-STATION, ETC.	JTSTP187
C*	RETURN FOR NEXT STEP IS MADE TO THIS POINT.	JTSTP188
C*		JTSTP189
	8 NSTP=NSTP+1	JTSTP190
	9 X=X+DX	JTSTP191
C*		JTSTP192
C*	IF COANNULAR PROBLEFM, SAVE UO,THO	JTSTP193
C*		JTSTP194
	IF(.NOT. TWD) GO TO 9966	JTSTP195
	USY0=UO	JTSTP196
	THSV0=THO	JTSTP197
9966	IF (MIX) CALL AITER1(X,DX)	JTSTP198
	NHALF=0	JTSTP199
	IF(RESTRI.NE.BITS) GO TO 10	JTSTP200

IF(ENTRY1 .OR. ICYCLE) GO TO 20	JTSTP201
C* IF 1-ST STFP PROPERTY CALCULATION IS BYPASSED	JTSTP202
C*	JTSTP203
C*	JTSTP204
10 CALL FMPYC(1,C6,Y,YD,NPU)	JTSTP205
CALL FMPYC(1,EJET,ED,TKE,NPU)	JTSTP206
CALL FMPYC(1,TJET,THD,T,NPU)	JTSTP207
C*	JTSTP208
C* CALL SCALE TO COMPUTE WIDTH OF MIXING ZONE(S)	JTSTP209
C* AND REFERENCE SCALES FOR TURBULENCE	JTSTP210
C*	JTSTP211
12 IF(SCLD) GO TO 13	JTSTP212
CALL SCALE(UU,TWOJ,NREG,X)	JTSTP213
GO TO 15	JTSTP214
13 CALL SCALE(ALXU(1,NC),TWOJ,NREG,X)	JTSTP215
C*	JTSTP216
C* COMPUTE PROPERTIES	JTSTP217
C*	JTSTP218
15 CALL PROPJ(TWOJ,TURBJ,NREG,X,YD,T,TKF,1,NPU)	JTSTP219
CALL GAMCP(T,GMC,CP,RG,1,NPU)	JTSTP220
GO TO LGOP. (16,1001)	JTSTP221
C*	JTSTP222
C* ADD MESH POINT TO DOWNSTREAM STATION IF ADDP=T.	JTSTP223
C*	JTSTP224
16 IF(.NOT.ADDP) GO TO 18	JTSTP225
17 CALL PADD(SM,NPD,NREG)	JTSTP226
ADDP=.FALSE.	JTSTP227
18 NM=NPD	JTSTP228
C*	JTSTP229
C* ASSURE THAT UPSTREAM PSI IS FAR ENOUGH OUT	JTSTP230
C*	JTSTP231
IF(SM1(NPU).EQ.SM(NPD)) GO TO 20	JTSTP232
SM1(NPU+1)=SM(NPD)	JTSTP233
C*	JTSTP234
C* EXTRAPOLATE APPROXIMATE Y	JTSTP235
C* LINEAR EXTRAPOLATION	JTSTP236
C*	JTSTP237
NPU=NPU+1	JTSTP238
NM1=NPU	JTSTP239
DYDPSI=(Y(NPU-1)-Y(NPU-2))/(SM1(NPU-1)-SM1(NPU-2))	JTSTP240
Y(NPU)=Y(NPU-1)+DYDPSI*(SM1(NPU)-SM1(NPU-1))	JTSTP241
UU(NPU)=UU(NPU-1)	JTSTP242
THU(NPU)=THU(NPU-1)	JTSTP243
EU(NPU)=EU(NPU-1)	JTSTP244
T(NPU)=TJET+THU(NPU)	JTSTP245
TKE(NPU)=EJET+EU(NPU)	JTSTP246
YD(NPU)=C6*Y(NPU)	JTSTP247
IF(.NOT. DIFF) GO TO 7002	JTSTP248
DO 7003 I=1,NC	JTSTP249
7003 ALXU(NPU,I)=ALXU(NPU-1,I)	JTSTP250

7002 CALL PROPJ(TWOJ,TURBJ,NREG,X,YD,T,TKF,NPU,NPU)	JTSTP251
CALL GAMCP(T,GMC,CP,RC,NPU,NPU)	JTSTP252
C*	JTSTP253
C*** SOLUTION OF MOMENTUM EQUATION	JTSTP254
C*	JTSTP255
20 CALL MOVE(I,UU,VAR,NM1,1)	JTSTP256
C*	JTSTP257
C* TEMPORARILY SAVE CURRENT Y VALUES ON ITERATION	JTSTP258
C*	JTSTP259
IF(MIX.AND.(.NOT.ICYCLE)) CALL MOVE(1,Y,YU,NPU,1)	JTSTP260
CALL SFTM(1,1,,EPS,NPU)	JTSTP261
CALL SETM(1,C1,AL,NPU)	JTSTP262
CALL SETM(2,0,,GM,NPU,DL,NPU)	JTSTP263
FNTRY1=.FALSE.	JTSTP264
IF(RFSTRT,NE.BITS) RFSTRT=BITS	JTSTP265
C*	JTSTP266
IF(.NOT.DPRIN) GO TO 19	JTSTP267
WRITE (6,8680) NSTP,X,DX	JTSTP268
8680 FORMAT(/,6X,5HSTEP=,14,3X,2HX=,1PE16,8,3X,3HDX=,E16,8//)	JTSTP269
WRITE (6,8681) YJEIF,SFEDGE,EDGE1,SF1	JTSTP270
8681 FORMAT(6X,6HYJEIF=,1PE16,8,6X,7HSFEDGE=,E16,8/,	JTSTP271
* 6X,6HEDGE1=,E16,8,6X,7HSF1 =,E16,8//)	JTSTP272
CALL TAPRT(NAM(1),Y,NPU,10,0)	JTSTP273
CALL TAPRT(NAM(2),SM,NPD,10,0)	JTSTP274
CALL TAPRT(NAM(3),XLN,NPD,10,0)	JTSTP275
19 DO 21 L=2,NPU	JTSTP276
RAD=1,	JTSTP277
IF(AXI) RAD=Y(1)**2	JTSTP278
IF(MIX)GM(1)=-C7*DPDX2/(RHO(L)*UU(L))	JTSTP279
23 BE(L)=MUFEFF(L)*RHO(L)*UU(L)*RAD	JTSTP280
IF(.NOT. TURBJ) BE(L)=MUL(L)*RHO(L)*UU(L)*RAD	JTSTP281
21 CONTINUE	JTSTP282
BF(1)=0,	JTSTP283
IF(.NOT. AXI) BE(1)=MUFEFF(1)*RHO(1)*UU(1)	JTSTP284
C*	JTSTP285
C* BOUNDARY CONDITIONS	JTSTP286
C*	JTSTP287
R1=0,	JTSTP288
C11=1,	JTSTP289
D1=0,	JTSTP290
AN=1,	JTSTP291
BN=0,	JTSTP292
22 DN=UEDGE	JTSTP293
IF(.NOT.MIX) GO TO 24	JTSTP294
C*	JTSTP295
C* B.C. FOR CONFINED MIXER	JTSTP296
C*	JTSTP297
IF(UPSTRM) GO TO 24	JTSTP298
AN=0,	JTSTP299
DN=GRADU	JTSTP300

C*	BN=1.	JTSTP301
C*	24 CALL DFER(0,0,0,IER)	JTSTP302
	IF(IER) GO TO 999	JTSTP303
C*	25 CALL MOVE(2,VAR,UD,NPD,1,DVAR,DUDY,NPD,1)	JTSTP304
	IF(UFDGE.EQ.0. .AND. UD(1).GT.1. .AND. (.NOT.SCLD)) UD(1)=1.	JTSTP305
C*		JTSTP306
C*		JTSTP307
C*		JTSTP308
C*		JTSTP309
C*		JTSTP310
C*	VALUES OF UD WITHIN UD(1)=UCL1, SET TO UCL1	JTSTP311
C*	UCL1=UD(1)	JTSTP312
	CLU=UCL1+1,E=6	JTSTP313
	CLL=UCL1-1,E=6	JTSTP314
100	DO 1111 I=1,NM	JTSTP315
	IF(UD(L).GE.CLL .AND. UD(L).LE.CLU) UD(L)=UCL1	JTSTP316
1111	CONTINUE	JTSTP317
	DO 1112 L=1,NPU	JTSTP318
	UK(L)=YDF(SM(L),SM,UD,1,NPD)	JTSTP319
	DUDY(L)=YDF(SM(L),SM,DVAR,1,NPD)	JTSTP320
1112	CONTINUE	JTSTP321
	IF(DPRIV) CALL TAPRT(NAM(4),UD ,NPD,10,0)	JTSTP322
C*		JTSTP323
C*		JTSTP324
C*		JTSTP325
C*	TEST FOR TURBULENT PROBLEM	JTSTP326
C*		JTSTP327
	28 IF(.NOT. TURBJ) GO TO 7010	JTSTP328
C*		JTSTP329
C****	SOLUTION OF THE EQUATION	JTSTP330
C*		JTSTP331
	30 NM1=NPU	JTSTP332
	CALL MOVE(1,EU,VAR,NPU,1)	JTSTP333
	DO 31 L=2,NPU	JTSTP334
	RAD=1.	JTSTP335
	IF(AXI) RAD=Y(L)**2	JTSTP336
	RD=RHO(L)*UK(L)	JTSTP337
	BE(L)=MUL(1)*DK(L)*RD*RD	JTSTP338
	DUDYSQ=DUDY(L)*DUDY(L)	JTSTP339
	GM(L)=C2*MUL(L)*(MUEFF(L)/MUL(L)-1.)*RD*DUDYSQ*RD	JTSTP340
	IF(RD.EQ.0. .OR. UD(L).LE..005) GO TO 33	JTSTP341
	DL(L)=C5*C*MUL(L)*DK(L)/(RD*XI*N(L)*XLN(L))	JTSTP342
	GO TO 31	JTSTP343
33	DL(L)=0.	JTSTP344
	GM(L)=0.	JTSTP345
31	CONTINUE	JTSTP346
	BE(1)=0.	JTSTP347
	IF(.NOT. AXI) BE(1)=MUL(1)*DK(1)*RHO(1)*UD(1)	JTSTP348
C*		JTSTP349
		JTSTP350

AN=1	JTSTP351
BN=0	JTSTP352
32 DN=EEDGE	JTSTP353
IF(.NOT. MIX) GO TO 34	JTSTP354
IF(UPSTRM) GO TO 34	JTSTP355
AN=0	JTSTP356
DN=0	JTSTP357
BN=1	JTSTP358
C*	JTSTP359
34 CALL DEEQ(0,0,0,IER)	JTSTP360
IF(IER) GO TO 999	JTSTP361
C*	JTSTP362
C* SCAN FOR POSSIBLE NEGATIVES AT JET EDGE--(UEEDGE=0)	JTSTP363
C*	JTSTP364
IF(UEEDGE.NF.0.) GO TO 35	JTSTP365
DO 3391 L=1,NPD	JTSTP366
IF(VAR(L).LT.0.) GO TO 3392	JTSTP367
3391 CONTINUE	JTSTP368
GO TO 35	JTSTP369
3392 LK=L-1	JTSTP370
LL=L	JTSTP371
SLOPE=(VAR(NPD)-VAR(LK))/(SM(NPD)-SM(LK))	JTSTP372
NPD1=NPD-1	JTSTP373
DO 3393 L=LL,NPD1	JTSTP374
3393 VAR(L)=VAR(L-1)+SLOPE*(SM(L)-SM(L-1))	JTSTP375
C*	JTSTP376
35 CALL MOVF(1,VAR,ED,NPD,1)	JTSTP377
CALL MOVE(2,DVAR,DIKE,NPD,1,BE,DSTOR(601),NPU,1)	JTSTP378
IF(DPRIN) CALL TABPRT(NAM(5),ED,NPD,10,0)	JTSTP379
C*	JTSTP380
C* TEST FOR SPECIES OFQS	JTSTP381
C*	JTSTP382
7010 IF(.NOT.DIFF) GO TO 50	JTSTP383
CALL SFTH(2,0,GM,NPU,01,NPU)	JTSTP384
C*	JTSTP385
C* SOLVE SPECIES EQUATIONS// AIR MOLE FRACTION	JTSTP386
C* COMPUTED BY DIFFERENCE(COMPONENT 1)	JTSTP387
C*	JTSTP388
7020 DO 7500 LL=2,NC	JTSTP389
NM1=NPU	JTSTP390
CALL MOVF(1,ALXU(1,LL),VAR,NPU,1)	JTSTP391
DO 7100 L=2,NPU	JTSTP392
RAD=1.	JTSTP393
IF(AXI) RAD=Y(L)**2	JTSTP394
RD=RHO(L)*UK(L)	JTSTP395
BE(L)=MU EFF(L)/SCM(LL)*RD*RAD	JTSTP396
7100 GM(L)=0.	JTSTP397
BE(1)=0.	JTSTP398
IF(.NOT.AXI) BE(1)=MU EFF(1)*RHO(1)*UN(1)/SCM(L1)	JTSTP399
C*	JTSTP400

C* BOUNDARY CONDITIONS	JTSTP401
C*	JTSTP402
AN=1	JTSTP403
BN=0	JTSTP404
7102 DN=ALENCE(1L)	JTSTP405
C*	JTSTP406
7110 CALL DFEQ(1,0,0,IER)	JTSTP407
IF(1ER) GO TO 999	JTSTP408
C*	JTSTP409
C* MOVE NEW MOLE FRACTIONS,ETC-INTERP. UPSTRM DALXU	JTSTP410
C*	JTSTP411
DO 7115 L=1,NPU	JTSTP412
IF(VAR(1),LT,0.) VAR(L)=0.	JTSTP413
7115 DALXU(L,1L)=YDF(SM1(L),SM,DVAR,1,NPD)	JTSTP414
CALL MOVE(1,VAR,ALX(1,LL),NPD,1)	JTSTP415
C*	JTSTP416
7500 CONTINUE	JTSTP417
C*	JTSTP418
C* COMPUTE AIR MOLE FRACTION AND UPSTREAM DERIVATIVE	JTSTP419
C*	JTSTP420
DO 7749 L=1,NPD	JTSTP421
ALX(L,1)=1	JTSTP422
DALXU(L,1)=0.	JTSTP423
DO 7750 LL=2,NL	JTSTP424
ALX(L,1)=ALX(L,1)-ALX(L,LL)	JTSTP425
DALXU(L,1)=DALXU(L,1)-DALXU(L,LL)	JTSTP426
7750 CONTINUE	JTSTP427
7749 CONTINUE	JTSTP428
C*	JTSTP429
IF(.NOT.DPRIN) GO TO 50	JTSTP430
DO 5092 1L=1,NC	JTSTP431
CALL TABPRT(CNAME(LL),ALX(1,LL),NPD,10,0)	JTSTP432
5092 CONTINUE	JTSTP433
C* TEST FOR HEAT TRANSFER EFFECTS	JTSTP434
C*	JTSTP435
C*	JTSTP436
50 IF(.NOT. QJET) GO TO 6554	JTSTP437
C*** SOLUTION OF ENERGY EQUATION	JTSTP438
C*	JTSTP439
51 NM1=NPU	JTSTP440
CALL SFTM(1,0,,DL,NPU,GM,NPU)	JTSTP441
IF(.NOT.TURBJ) GO TO 54	JTSTP442
C*	JTSTP443
C*	JTSTP444
C* COMPUTE D(OTKE/DY)/DY	JTSTP445
C*	JTSTP446
C* COMPUTE TERMS ENTERING INTO SOURCE	JTSTP447
C*	JTSTP448
C*	JTSTP449
DO 52 L=1,NPU	JTSTP450

BE(L)=C3*DSTDR(L+600)*YOF(SM1(L),SM,DTKE,1,NPD)	JTSTP451
52 CONTINUE	JTSTP452
NPM=NPU+1	JTSTP453
53 CALL DERIV,SM1,BE,GM,2,NPM	JTSTP454
C*	JTSTP455
54 DO 55 L=2,NPU	JTSTP456
57 RAD=1.	JTSTP457
IF(AXI) RAD=Y(L)**2	JTSTP458
RD=RHO(L)*UK(L)	JTSTP459
AL(L)=AL(L)/CP(L)	JTSTP460
BE(L)=KCP(1)*CP(L)*RD*RD	JTSTP461
DUDYSQ=DUDY(L)*DUDY(L)	JTSTP462
GM(L)=GM(L)/CP(L)+C4*RD*RD*MUEFF(1)*DUDYSQ/CP(L)	JTSTP463
* -(YOF(SM1(L),SM,ED,1,NPD)-EU(L))/(DX*CP(L))*EJET/TJET	JTSTP464
58 IF(.NOT. DIFF) GO TO 55	JTSTP465
ETA(L)=TJET*END(1)*MUEFF(1)*RD*RD/CP(L)*SUMCPD(1)	JTSTP466
55 CONTINUE	JTSTP467
BE(1)=0.	JTSTP468
IF(.NOT. AXI) BE(1)=KCP(1)*CP(1)*RHO(1)*UK(1)	JTSTP469
C*	JTSTP470
CALL MOVE(1,THU,VAR,NPU,1)	JTSTP471
DN=THEDGE	JTSTP472
IF(.NOT. MIX) GO TO 56	JTSTP473
C*	JTSTP474
C* B.C. FOR CONFINED MIXER	JTSTP475
C*	JTSTP476
IF(UPSTRM) GO TO 56	JTSTP477
AN=0.	JTSTP478
BN=1.	JTSTP479
DN=0.	JTSTP480
56 CALL DFEQ(0,0,0,IER)	JTSTP481
IF(IER) GO TO 999	JTSTP482
IF(DIFF) CALL SETM(1,0.,ETA,200)	JTSTP483
C*	JTSTP484
IF(UEGGE .NE. 0.) GO TO 60	JTSTP485
DO 6691 L=1,NPD	JTSTP486
IF(VAR(L).LT. 0.) GO TO 6692	JTSTP487
6691 CONTINUE	JTSTP488
GO TO 60	JTSTP489
6692 LK=L-1	JTSTP490
LL=L	JTSTP491
SLOPE=(VAR(NPD)-VAR(LK))/(SM(NPD)-SM(LK))	JTSTP492
NPD1=NPD-1	JTSTP493
DO 6693 L=1,NPD1	JTSTP494
6693 VAR(L)=VAR(L-1)+SLOPE*(SM(L)-SM(L-1))	JTSTP495
60 CALL MOVE(1,VAR,THD,NPD,1)	JTSTP496
IF(DPRIN) CALL TABPRI(NAM(6),THD,NPD,10,0)	JTSTP497
C*	JTSTP498
C*	JTSTP499
C* COANNULAR PROBLEM-- IF .NOT. MERGE. COMPUTE UO	JTSTP500

C*		JTSTP501
6554	IF(.NOT. TWO) GO TO 6555	JTSTP502
	IF(MERGE.OR. (.NOT. MIX)) GO TO 6555	JTSTP503
	IF(MERGE) GO TO 6555	JTSTP504
	UO=SQRT(USV0**2=144.*GC*RG*TJET*THSV0*(P2-P1)/P1)	JTSTP505
6555	CONTINUE	JTSTP506
C*		JTSTP507
C*		JTSTP508
C*		JTSTP509
C*		JTSTP510
C*	LOCATE EDGE OF JET--- ADD POINT IF NECESSARY	JTSTP511
C*		JTSTP512
86	CALL JTEDGE(X,YJETE,SFEDGE,ADDP)	JTSTP513
C*		JTSTP514
87	CONTINUE	JTSTP515
C*		JTSTP516
C*	CONFINED MIXER-CHECK FOR CONVERGENCE	JTSTP517
C*	OF PRESSURE ITERATION	JTSTP518
C*		JTSTP519
C*		JTSTP520
	IF(.NOT. MIX) GO TO 80	JTSTP521
	CALL ATTER2	JTSTP522
	IF(ADDP .AND. (.NOT. UPSTRM)) ADDP=FALSE	JTSTP523
	IF(ERR) RETURN	JTSTP524
	IF(CVG) GO TO 80	JTSTP525
	NMI=NPU	JTSTP526
	CALL MOVE(1,YU,Y,NPU,1)	JTSTP527
	ICYCLE=TRUE	JTSTP528
	IF(.NOT. TWO) GO TO 9966	JTSTP529
C*		JTSTP530
C*	RESTORE ON ITERATION	JTSTP531
C*		JTSTP532
	UO=USV0	JTSTP533
	THO=THSV0	JTSTP534
	GO TO 9966	JTSTP535
C*		JTSTP536
C*	IF UPSTRM=F, SET SM(NM)=PSID FOR CONFINED MIXER	JTSTP537
C*	NO MESH POINTS WILL BE ADDED AFTER THIS POINT	JTSTP538
C*		JTSTP539
80	IF (MIX .AND. (.NOT. UPSTRM)) SM(NM)=PSID	JTSTP540
C*		JTSTP541
C*	MOVE DOWNSTREAM CO-ORDINATES TO UPSTREAM TABLE	JTSTP542
C*		JTSTP543
C*		JTSTP544
	CALL MOVE(1,SM,SM1,NM,1)	JTSTP545
	ICYCLE=FALSE	JTSTP546
C*		JTSTP547
C*	TWO- JET LOGIC TO TEST FOR INTERACTION OF INNER AND OUTER JETS	JTSTP548
C*		JTSTP549
	IF((.NOT. TWO) .OR. (.NOT. MERGE) .OR. MFRSTP) GO TO 776	JTSTP550

C*		JTSTP551
C*	JETS HAVE MERGED	JTSTP552
C*		JTSTP553
	MER=.TRUE.	JTSTP554
	MERSTP=.TRUE.	JTSTP555
	XMRG=XMERGE	JTSTP556
	GO TO 102	JTSTP557
C*		JTSTP558
C*		JTSTP559
C*	TEST FOR DISAPPEARANCE OF SUPERSONIC CORE IF JET IS SUPERSONIC	JTSTP560
C*		JTSTP561
	776 IF(SUBSON) GO TO 777	JTSTP562
	TCL=TJET*THD(1)	JTSTP563
	VCL=VJET*UD(1)	JTSTP564
	VSONC = SQRT(GMC(1)*GC*RG*TCL)	JTSTP565
	IF(VCL.GE.VSONC) GO TO 777	JTSTP566
C*		JTSTP567
C*	SUPERSONIC CORE HAS JUST DISAPPEARED	JTSTP568
C*		JTSTP569
	SUPC=.TRUE.	JTSTP570
	SUPSTP=.TRUE.	JTSTP571
C*		JTSTP572
C*	FLAG NOW SUBSONIC JET	JTSTP573
C*		JTSTP574
	SUBSON=.TRUE.	JTSTP575
	XSUP=X	JTSTP576
	GO TO 102	JTSTP577
C*		JTSTP578
C*		JTSTP579
C*	** SEQUENCE OF TESTS FOR DISAPPEARANCE OF THE	JTSTP580
C*	** POTENTIAL CORE OR THE LAST STEP	JTSTP581
	777 IF(LAST) GO TO 220	JTSTP582
	IF(SCLD) GO TO 778	JTSTP583
	IF(UD(2),EQ,UD(3)) GO TO 310	JTSTP584
	GO TO 779	JTSTP585
	778 IF(ALX(1,NC).LE.ALXLM) GO TO 310	JTSTP586
	779 IF(CORSTP) GO TO 310	JTSTP587
C*		JTSTP588
C*	CORE HAS JUST DISAPPEARED	JTSTP589
C*		JTSTP590
	CORE=.TRUE.	JTSTP591
101	CORSTP=.TRUE.	JTSTP592
	NREG=2	JTSTP593
	XLC=X	JTSTP594
	XCORF=X	JTSTP595
102	CALL XSIZE(IDX,X,REFL,NREG,LAST)	JTSTP596
	GO TO 500	JTSTP597
C*		JTSTP598
C*		JTSTP599
C*	TEST FOR DISAPPEARANCE OF CORE	JTSTP600

C*	220 LAST=.FALSE.	JTSTP601
	IF(SCLD) GO TO 224	JTSTP602
	IF((UD(2),FD,UD(3)) .OR. CORSTP) GO TO 500	JTSTP603
	GO TO 101	JTSTP604
C*		JTSTP605
C*	CORE HAS JUST DISAPPEARED-SET CORSTP=1	JTSTP606
C*		JTSTP607
	224 IF(ALX(1,NC),LE,ALXLIM .OR. CORSTP) GO TO 500	JTSTP608
	GO TO 101	JTSTP609
C*		JTSTP610
C*		JTSTP611
C*	TEST FOR NO. OF MESH POINTS,GT,NM	JTSTP612
C*		JTSTP613
	310 IF(NPD.GT. NMAX) CALL MSHCUT(NREG,SH,NPD)	JTSTP614
C*		JTSTP615
C*	ADJUST X-STEP SIZE	JTSTP616
C*		JTSTP617
	320 CALL XSIZE(DX,X,REFL,NREG,LAST)	JTSTP618
	CALL DTEST	JTSTP619
	IF(ERR) GO TO 500	JTSTP620
C*		JTSTP621
	CALL MOVE(3,UD,UU,NM1,1,ED,EU,NM1,1,THD,THU,NM1,1)	JTSTP622
	NPU=NM1	JTSTP623
	IF(.NOT.DIFF) GO TO 8	JTSTP624
	DO 7700 LL=1,NC	JTSTP625
	CALL MOVE(1,ALX(1,LL),ALXU(1,LL),NM1,1)	JTSTP626
	7700 CONTINUE	JTSTP627
	GO TO 8	JTSTP628
C*		JTSTP629
C*	ERROR RETURN	JTSTP630
C*		JTSTP631
	999 ERR=.TRUE.	JTSTP632
C*		JTSTP633
	500 CONTINUE	JTSTP634
	NPU=NM1	JTSTP635
	1000 CALL MOVE(4,UD,UU,NM1,1,ED,EU,NM1,1,THD,THU,NM1,1,	JTSTP636
	* SM1,PSI,NM1,1)	JTSTP637
	IF(.NOT.DIFF) GO TO 7740	JTSTP638
	DO 7730 LL=1,NC	JTSTP639
	CALL MOVE(1,ALX(1,LL),ALXU(1,LL),NM1,1)	JTSTP640
	7730 CONTINUE	JTSTP641
	7740 ASSIGN 1001 TO LGOP	JTSTP642
	GO TO 10	JTSTP643
C*		JTSTP644
	1001 IF(PM(10).NE. 0.) CALL TAPRT(NAM(3),XLN,NPD,10.0)	JTSTP645
	RETURN	JTSTP646
	END	JTSTP647
		JTSTP648

CKINET	HYDROCARBON KINETICS ROUTINE	KINET001
	SUBROUTINE KINET(P,H,FOA1,DT)	KINET002
C		KINET003
	CHEM(X,Y)= (X*((X+1,-BETA0)*EXP(Y)-(X-1,+BETA0)))/	KINET004
*	((X+1,-BETA0)*EXP(Y)+(X-1,+BETA0))	KINET005
	LOGICAL RATECO	KINET006
	LOGICAL MOLIN	KINET007
	COMMON /CBITS / BITS, BLANK	KINET008
	COMMON /PSEQ / FDA,BETA,TP,X(16),DHQDM,TEQ,BEQ,XMWT,HNEQ	KINET009
	EQUIVALENCE (X1,X(1)),(X2,X(2)),(X3,X(3)),(X4,X(4)),	KINET010
*	(X5,X(5)),(X6,X(6)),(X7,X(7)),(X8,X(8)),	KINET011
*	(X9,X(9)),(X10,X(10)),(X11,X(11))	KINET012
	COMMON /PRQPR / PR,HHR,TR,FCR,RHOR,RR,MMWTR	KINET013
	COMMON /CKINET/ TIME,TIME,NTSTEP,DBFT,XMOLNO,CRC,NREAC,	KINET014
*	RC1(9),RC2(9),RC3(9),TBE(11,5),RCON(5),	KINET015
*	TK,ENTRY1	KINET016
	LOGICAL CRC,ENTRY1	KINET017
	EQUIVALENCE (RK1,RCON(1)),(RK2,RCON(2)),(RK3,RCON(3)),	KINET018
*	(RK4,RCON(4)),(RK5,RCON(5))	KINET019
	COMMON /KININS/ XMWUN,XMWC,TCNST,CNER,XNOI	KINET020
	LOGICAL TCNST,CNER	KINET021
	LOGICAL FIRST, LAST	KINET022
	COMMON /COIREM/ YTOL,DUM(3)	KINET023
	COMMON /CPRINT/ PDUM(20)	KINET024
	COMMON /INMOLF/ XIN(12)	KINET025
	COMMON /CRATE / RATE	KINET026
	COMMON /GHSC / FF(25),HZ(25),SR(25),CPZ(25),DCPR(25)	KINET027
	COMMON /CEQKIN/ BK,A1K,DELHNO,DEN,XMWEQ,XNO,XCN	KINET028
	COMMON /CCDCO2/ CORATE,XCOI	KINET029
	LOGICAL CORATE	KINET030
	COMMON /TROUBL/ FRR,ERRMAJ,INERR,PRERR	KINET031
	LOGICAL FRR,ERRMAJ,INERR,PRERR	KINET032
	COMMON /PSEQX / HOC,HUM,CO2AIR,MAIR,FS,FUELMW	KINET033
	REAL MAIR	KINET034
	COMMON /CPSEQ2/ SP4,SPMU,SPMC	KINET035
	COMMON /DPHETA/ BETADP	KINET036
	DOUBLE PRECISION BETADP	KINET037
	COMMON /SNM / ALSP(150), MMTC(75)	KINET038
	COMMON /COLIMT/ XCOLIM	KINET039
	DIMENSION QV(8)	KINET040
	DIMENSION XSAV(12)	KINET041
C		KINET042
	DATA C2/.01603286/	KINET043
	DATA H0/1.98596/	KINET044
	NAMLIST /A/ ER1,ERSAV,BETA,BETS,ENTRY1,IK,TIME,	KINET045
*	DTIME,TP,XMW1,XMWT,XMWC,XMWUN,X,A1,RATE,XMWEQ,XNO	KINET046
*	,DHQDM,DELH,HHR,RCON,BETA0,BK,XMWEQ,DELHNO,CPM	KINET047
*	,BEQ,TEQ	KINET048
*	,QV,TP1,HM,MH1,ERRT,BETADP	KINET049
C		KINET050

C	**REACTIONS**	(1)	H	+OH	+M	=	H2O+M	KINET051
C		(2)	H	+H	+M	=	H2+M	KINET052
C		(3)	H	+O	+M	=	OH+M	KINET053
C		(4)	O	+O	+M	=	O2+M	KINET054
C		(5)	H	+O2	+M	=	HO2+M	KINET055
C		(6)	H	+HO2		=	OH+OH	KINET056
C		(7)	OH	+HO2		=	H2O+O2	KINET057
C		(8)	O	+HO2		=	OH+O2	KINET058
C		(9)	H	+HO2		=	H2+O2	KINET059
C								KINET060
C								KINET061
C								KINET062
	IADD = 0							KINET063
	IF(CONER .AND. (.NOT.ENTRY1)) GO TO 20							KINET064
	IF(.NOT.ENTRY1) GO TO 1							KINET065
	FUELMW= 12.01+1.008*HQC							KINET066
	FS = FUELMW/(1+.25*HQC)*.209495/MAIR							KINET067
1	FOA = FOA1							KINET068
	ER1 = FOA/FS							KINET069
	HH = H							KINET070
	IF(ERSV.EQ.0.) GO TO 12							KINET071
C								KINET072
C	ADJUST BETA, COMPOSITIONS FOR ADDED AIR							KINET073
C								KINET074
	F1 = ERSV*FS							KINET075
	F2 = FOA1							KINET076
	FB1 = F1/(1.+F1+HUM)							KINET077
	FB2 = F2/(1.+F2+HUM)							KINET078
	DEL = FB1/FB2-1.							KINET079
	XMSAV = XMW1							KINET080
	TERMW = DEL*(1./MAIR+HUM/XMTC(6))/(1.+HUM)							KINET081
	XMW1 = (1.+DEL)/(1./XMSAV+TERMW)							KINET082
	TERMW1= XMW1/(1.+DEL)							KINET083
	TERMP1= DEL/(1.+HUM)							KINET084
	TERMP2= XMW1/(XMSAV*(1.+DEL))							KINET085
	CALL FMPYC(1,TERMP2,X,X,3)							KINET086
	X5 = TERMP2*X5							KINET087
	X7 = TERMP2*X7							KINET088
	X11 = TERMP2*X11							KINET089
	X4 = TERMW1*(X4/XMSAV+TERMP1*.209495/MAIR)							KINET090
	X6 = TERMW1*(X6/XMSAV+TERMP1*HUM/XMTC(6))							KINET091
	X8 = TERMW1*(X8/XMSAV+TERMP1*CO2AIR/MAIR)							KINET092
	X9 = TERMW1*(X9/XMSAV+TERMP1*.780881/MAIR)							KINET093
	X10 = TERMW1*(X10/XMSAV+TERMP1*(.009624-CO2AIR)/MAIR)							KINET094
	TMOLES= 0.							KINET095
	DO 5 I=1,11							KINET096
5	TMOLES= TMOLES+X(I)							KINET097
	RADDM = 1./TMOLES							KINET098
	CALL FMPYC(1,RADDM,X,X,12)							KINET099
	XNO = X(11)							KINET100

	XCO = X7	KINET101
	IADD = 1	KINET102
C		KINET103
C		KINET104
C	IF FIRST ENTRY-- INITIALIZE COMPOSITIONS AT XMOLW0	KINET105
C		KINET106
	12 IF(.NOT. ENTRY1) GO TO 20	KINET107
	DHQDMW= 300,	KINET108
	XNO = XNOI	KINET109
	XCO = XCOI	KINET110
	TIME = 0.	KINET111
	FRSAV = 0.	KINET112
	MOLIN = .FALSE.	KINET113
	CALL SE1M(1,0,,X,11)	KINET114
	X11 = XNOI*X3	KINET115
	X7 = XCOI*X3	KINET116
	IF(XMOLW0.NE.BITS) GO TO 15	KINET117
	MOLIN = .TRUE.	KINET118
	XMOLW0= 0.	KINET119
	DO 13 K=1,12	KINET120
	13 XMOLW0= XMOLW0+XIN(K)*WMTC(K)	KINET121
	XNO = XIN(11)	KINET122
	XCO = XIN(7)	KINET123
	CALL MOVE(1,XIN,X,12,1)	KINET124
	15 CALL EQKIN(P,HH,,FALSE.)	KINET125
	BETA = (1./XMOLW0-1./XHWUN)*DEN	KINET126
	BETA0 = BETA	KINET127
	16 BETADP= BETA	KINET128
	IF(MOLIN) GO TO 17	KINET129
	X7 = XCO	KINET130
	X11 = XNO	KINET131
	CALL PSEQ2(FDA,HUM,HOC,TP,BETADP,CORATE,,TRUE,,X)	KINET132
	GO TO 18	KINET133
	17 CALL MOVE(1,XIN,X,12,1)	KINET134
	18 XNO = X11	KINET135
	XCO = X7	KINET136
	XMW1 = XMOLW0	KINET137
	XWMT = XMOLW0	KINET138
	RHO = 144.*XWMT*P/(1545.32*TP)	KINET139
	IF(TCONST) HNEQ=BITS	KINET140
	ENTRY1= .FALSE.	KINET141
	RETURN	KINET142
C		KINET143
C	INTEGRATE RATE EQUATIONS OVER GIVEN TIME STEP	KINET144
C		KINET145
C		KINET146
C	ESTIMATE NUMBER OF TIME STEPS AT ERI	KINET147
C		KINET148
	20 IF(N1STEP.EQ.0) GO TO 291	KINET149
	DTIME = DT/FLQAT(N1STEP)	KINET150

IF(BK.EQ.0.) IK=1	KINET151
GO TO 223	KINET152
291 IF(BETA.GE.1.) GO TO 222	KINET153
IF(BK.NE.0.) GO TO 21	KINET154
IK = 0	KINET155
ASSIGN 201 TO LGO.	KINET156
GO TO 24	KINET157
201 DTIMAX= DBET/A1	KINET158
IK = 1	KINET159
GO TO 222	KINET160
21 ASSIGN 22 TO LGO	KINET161
GO TO 24	KINET162
22 DTIMAX= DBET/(A1*(1.-BETA)**2-BK)	KINET163
222 DTIME = AMIN(DT,DTIMAX)	KINET164
223 TIME2 = TIME+DT	KINET165
ASSIGN 32 TO LGO	KINET166
LAST = .FALSE.	KINET167
TIME = TIME+DTIME	KINET168
IF(TIME.LT.TIME2) GO TO 241	KINET169
LAST = .TRUE.	KINET170
241 CALL MOVE(1,X,XSAV,12,1)	KINET171
IF(TCONST) GO TO 245	KINET172
CALL EQKIN(P,HH,.FALSE.)	KINET173
XCOLIM= X7	KINET174
RHO = 144.*XMW1*P/(1545.32*TP)	KINET175
IF(ER1.LT.PDUM(6).AND.ER1.GE.PDUM(5)) WRITE (6,A)	KINET176
RATECO= CORATE	KINET177
245 CALL MOVE(1,XSAV,X,12,1)	KINET178
IF(CUNER) GO TO 24	KINET179
BETA = (1./XMW1-1./XMWUN)*DEN	KINET180
IF(BETA.GE.BEQ) BETA=BEQ	KINET181
IF(TP.GE.TEQ) TP=TEQ	KINET182
BETA0 = BETA	KINET183
IF(IADD.EQ.0) GO TO 24	KINET184
C	KINET185
C* RECALCULATE PSEUDO-EQUILIBRIUM AFTER DILUTION	KINET186
C	KINET187
BETAOP= BETA	KINET188
QV(1) = 0.	KINET189
TP1 = TP	KINET190
XJP = .005*TP	KINET191
2451 CALL PSE22(FUA,HUM,HQC,TP1,BETAOP,RATECO,.TRUE.,X)	KINET192
HH1 = 0.	KINET193
DO 2452 K=1,11	KINET194
2452 HH1 = HH1+X(K)*HZ(K)	KINET195
HH1 = RO*TP1*HH1/XMW1	KINET196
ERRT = HH-HH1	KINET197
CALL DIHEM(TP1,ERRT,XJP,QV)	KINET198
IF(QV(1).GE.80.) WRITE (6,36) TP,TP1,ERRT,HH,HH1,X	KINET199
IF(QV(1).NE.0.) GO TO 2451	KINET200

	TP = TP1	KINET201
	IADD = 0	KINET202
C		KINET203
C		KINET204
C	IF CD RATE=1, INTEGRATE CO RATE EQUATION	KINET205
C		KINET206
	24 IF(.NOT. RATECO) GO TO 2460	KINET207
	XCOSAV= X7	KINET208
	CALL COCO2R(P, DTIME)	KINET209
	XCO = X7	KINET210
	DXCO = XCO-XCOSAV	KINET211
	X6 = X6+DXCO	KINET212
	X3 = X3+DXCO	KINET213
C		KINET214
C	CALCULATE FULL EQUILIBRIUM WITH FIXED CO, NO	KINET215
C		KINET216
	CALL MOVE(1,X,XSAV,12,1)	KINET217
	CALL EQKIN(P, HH, .TRUE.)	KINET218
	CALL MOVE(1,XSAV,X,12,1)	KINET219
	2460 IF(ER1, LI, PDUM(6), AND, ER1, GE, PDUM(5)) WRITE (6,A)	KINET220
	IF(BETA, EQ, REQ) GO TO 40	KINET221
	TK = TP/1.8	KINET222
	RHOCGS= C2*RHO	KINET223
	25 CALL RATCON(P)	KINET224
	30 RATE = -RHOCGS**2/ XMW1**3*(RK1*X1*X5+RK2*X1**2+RK3*X1*X2	KINET225
	* +RK4*X2**2+RK5*X1*X4)	KINET226
	IF(RATE, EQ, 0.) GO TO 33	KINET227
	A1 = RATE/((1./XMW1-1./XMW2)**2*DEN)	KINET228
	IF(RK, NE, 0.) GO TO 31	KINET229
	IF(TK, EQ, 0) GO TO 301	KINET230
C		KINET231
C	LIMITING BETA AS RK GOES TO 0	KINET232
	BETA = 1./((1.-BETA0)/(1.+(1.-BETA0)*A1*DTIME)	KINET233
	GO TO 33	KINET234
C	31 AA = SQRT(BK/A1)	KINET235
C	AAA = 2.*SQRT(BK*AA)	KINET236
C	***** MODS FOR TEST DECK *****	KINET237
	31 AA = 1./BEQ	KINET238
	AAA = 2.*SQRT((A1*AA)**2)	KINET239
C	*****	KINET240
	301 GO TO LGJ , (22,32,201)	KINET241
	32 AAA = AAA*DTIME	KINET242
	BETA = 1.-CHEM(AA,AAA)	KINET243
	33 XMW1 = 1./((BETA/DEN+1./XMWUN)	KINET244
	IF(BETA, GE, REQ) BETA=BEQ	KINET245
	IF(BETA, EQ, REQ) XMW1=XMWEO	KINET246
	BETA0 = BETA	KINET247
	IF(TCOVST) GO TO 40	KINET248
C		KINET249
C		KINET250

C	ITERATE FOR TEMPERATURE	KINET251
C		KINET252
	BETADP= BETA	KINET253
	QV(1) = 0.	KINET254
	TP1 = TP	KINET255
	XJP = .01*TP	KINET256
34	CALL PSEQ2(FOA,HUM,HOC,TP1,BETADP,RATECO,,TRUE,,X)	KINET257
	HH1 = 0.	KINET258
	DO 35 K=1,11	KINET259
35	HH1 = HH1+X(K)*HZ(K)	KINET260
	HH1 = R0*TP1*HH1/XMWT	KINET261
	ERRT = HH-HH1	KINET262
	CALL JIRFM(TP1,ERRT,XJP,QV)	KINET263
	IF(QV(1),GE.80.) WRITE (6,36) TP,TP1,ERRT,HH,HH1,X	KINET264
36	FORMAT(/2X,39HKINET I ITERATION TP,TP1,ERRT,HH,HH1,X/	KINET265
	* 1X,5E16.8/1X,6E16.8/1X,6E16.8)	KINET266
	IF(QV(1),NE.0.) GO TO 34	KINET267
	TP = TP1	KINET268
C		KINET269
40	XND = X11	KINET270
	XMW1 = XMWT	KINET271
	RHO = 144.*XMWT*P/(1545.32*TP)	KINET272
C		KINET273
C	PREDICT NO AT END OF TIME STEP	KINET274
401	XNOSAV= XND	KINET275
C		KINET276
	CALL NOX2B(P,DTIME,XND)	KINET277
C		KINET278
C	ADJUST MOLE FRACTIONS FOR NO== N ATOMS NOT USED	KINET279
	DXND = XND-XNOSAV	KINET280
	X4 = X4-.5*DXND	KINET281
	X9 = X9-.5*DXND	KINET282
	X11 = XND	KINET283
99	IF(LAST) GO TO 100	KINET284
	TIME = TIME+DTIME	KINET285
	IF(TIME,LT,TIME2) GO TO 24	KINET286
	LAST = .TRUE.	KINET287
	TIME = AMIN1(TIME,TIME2)	KINET288
	GO TO 24	KINET289
100	ERSAV = ER1	KINET290
	TIME = TIME2	KINET291
	RETURN	KINET292
	END	KINET293

C	KINPP	READ INPUT DATA AND INITIALIZE	KINPP001
C		SPECIAL VERSION - GE-AEG	KINPP002
C		8/23/73 ** MODIFICATIONS FOR GCKP-1 **	KINPP003
C		SUBROUTINE KINP	KINPP004
C		INPUT CAN BE ACCEPTED IN (1) INTERNAL (CGS) UNITS, (2) FPS UNITS,	KINPP005
C		(3) SI UNITS	KINPP006
C		THE FOLLOWING UNITS ARE USED INTERNALLY	KINPP007
C		* DISTANCE CM *	KINPP008
C		* AREA CM**2 *	KINPP009
C		* MASS FLOW RATE GM/SEC *	KINPP010
C		* PRESSURE ATM *	KINPP011
C		* TIME SEC *	KINPP012
C		* VELOCITY CM/SEC *	KINPP013
C		* DENSITY GM/CC *	KINPP014
C		* TEMPERATURE DEG K *	KINPP015
C		* CONCENTRATION MOLE(I)/MASS *	KINPP016
C		INTERNAL CORRESPONDENCE	KINPP017
C		* DVAR = DEPENDENT VARIABLE *	KINPP018
C		* IVAR = INDEPENDENT VARIABLE *	KINPP019
C		* AVAR = ASSIGNED VARIABLE *	KINPP020
C		THE FOLLOWING LOGICAL TAPE UNITS ARE REQUIRED	KINPP021
C		* LTHM (4) = FOR THERMODYNAMIC DATA *	KINPP022
C		* LDAT (43) = FOR TEMPORARY STORAGE OF DATA CARDS *	KINPP023
C		* 03 = FOR OPTIONAL BINARY OUTPUT FILE *	KINPP024
C		LOGICAL TAPE UNIT ASSIGNMENTS ARE SPECIFIED IN 'NAMBLK'	KINPP025
C		DOUBLE PRECISION DSP,DSPP,DSPNM,DAISP	KINPP026
C		LOGICAL ALIM1,CONC,DBUGO,ELIM,FVSTEP,EXCHR,MOLFF,MMHG	KINPP027
C		LOGICAL COMBIS,RHOCN,SHOCK,TCN	KINPP028
C		LOGICAL NEWPRT,NEXT,BRIEF	KINPP029
C		REAL MODT,IVAR,M,MM,N,LSUBM,MIXMM,M2,NEW	KINPP030
C		DIMENSION ISS(25),IBR(3),CXIB(40),CAIB(40),APRINT(50),THMC(7,2)	KINPP031
C		DIMENSION SP(2,4),DSP(4),SPP(2,3),DSPP(3),SPNM(2,27),DSPNM(27)	KINPP032
C		DIMENSION IPT(2),LMT(4),SUBS(4),C(25),CX(4)	KINPP033
C		DIMENSION CUA(2),FUA(2),SUA(2),CUP1(2),CUP2(2),FUP(2),SUP(2)	KINPP034
C		DIMENSION HFAD(10)	KINPP035
C		COMMON/LTUS/LTHM,LDAT	KINPP036
C		COMMON/PTG/VERST,TIMEV,VERSA,AREAV,ELIM,TCN,RHOCN,IPRCD	KINPP037
C		COMMON/CNND/DVAR,AREA,MODT,P,IVAR,V,P4D,T,SIGMA(25),LS,LSPT3,NEXT	KINPP038
C		COMMON/RFAC/LSR(4,30),XX(30),RATE(30),LKEQ(30),DLKEQ(30),MM(30),LR	KINPP039
C		COMMON/RRAT/A(30),N(30),FACT(30),B(30),M(25,30),ALIM1	KINPP040
C		COMMON/AFUN/CN(4),ITPSZ,LSUBM,FTA,D,VISC,BFTA	KINPP041
C			KINPP042
C			KINPP043
C			KINPP044
C			KINPP045
C			KINPP046
C			KINPP047
C			KINPP048
C			KINPP049
C			KINPP050

COMMON/SPEC/SMAX(2,30),MW(25),W(25),STOIC(25,30),OMEGA(25,30)	KINPP051
COMMON/STNT/HMTN,HI,MN,HNP1,HMAX,NH,AVH,EMAX,ERRN,ICV,KOINT,FRRP	KINPP052
COMMON/TCDF/TC(7,2,25),TLOW,TMID,THI	KINPP053
COMMON/PRIN/PRINT(50),NP,CEND,EVSTFP	KINPP054
COMMON/XVSA/XTB(40),ATB(40),NT,XU,AU(2),CX3,CX2,CX1,CX0	KINPP055
COMMON/XVSA/BRIEF,TIMLMT	KINPP056
COMMON/SNMM/DALSP(75),ALMW(75)	KINPP057
COMMON/KOUT/TITLE(20),UNITT,UNITO,CONC,EXCHR,DELH(10),FPS,SI,DBUGO	KINPP058
COMMON/GHSP/GRT(25),HRT(25),SR(25),CPR(25),DCPR(25)	KINPP059
COMMON/PORF/PK(28),OK(28),RK(28),E(28)	KINPP060
COMMON/SKIP/NEGL(25),T1,T2,IT	KINPP061
COMMON/VFCF/RR,MIXM,42,GAMMA,TCPR,R	KINPP062
COMMON/MISF/IT,PP,CPR0,HPO,ENN,SUMN,FNNL,LLMT(15),RO(15)	KINPP063
COMMON/INDX/TP,HP,NLM,NS,IQ1,CONVG,KMAT,THAT	KINPP064
COMMON / LSLRSV / LSSV, LRSV	KINPP065
COMMON / RINSAV / HMINZ,HMAXZ,HINTZ,FMAXZ,PRINTZ(50),NPRNTZ,	KINPP066
EVSTP7,DRUGOZ	KINPP067
LOGICAL EVSTP7,DRUGOZ	KINPP068
COMMON / STCTRL / DUMST(4),FIRSTC,DUMST1(12)	KINPP069
LOGICAL FIRSTC	KINPP070
EQUIVALENC (C,SIGMA),(SPNM,DSPNM),(NSP,SP),(DSPP,SPP),(SPT,SP)	KINPP071
EQUIVALENC (SPNM,SNM(1,4)),(FFFH,SPNM(1,26)),(BLANK,SPNM(1,27))	KINPP072
EQUIVALENC (FX3,CX),(CEND,END)	KINPP073
DATA CU,FU,SU/2HCH,2HFT,2HM /	KINPP074
DATA CUA/4HCH**1H2/,FUA/4HFT**1H2/,SUA/4HM**2,1H /	KINPP075
DATA CUP1/4HMHG,1H /,CUP2/3HATM,1H /,FUP/4HIB/F,4HT**2/,SUP/4HM/M	KINPP076
2H2/	KINPP077
DATA NFN,CHANGE,REPEAT/3HNEW,4HCHAN,4HREPE/	KINPP078
DATA TAPEND,CARDS/3HEND,4HCARD/	KINPP079
DATA HEAD(1)/60H ***** GCKP=1 INPUT *****	KINPP080
NAMELIST/PROB/HMIN,HMAX,HINT,EMAX,ALLM1,EIIM,CONC,EXCHR,	KINPP081
* IPRCON,ITPSZ,XTB,ATR,NTB,CX3,CX2,CX1,CX0,LSURM,ETA,D,VISC,RFTA	KINPP082
* END,DELP,PRINT,NPRNTS,APRINT,EVSTEP,DBUGO,	KINPP083
* COMBUS,SHOCK,TCOV,RHCON,BRIEF,TIMLMT	KINPP084
THERMODYNAMIC DATA WILL BE INPUT FROM BLOCK DATA -BIKTH-	KINPP085
ACTION = NFD	KINPP086
GO TO 4	KINPP087
ENTRY RINP	KINPP088
NEXT = .FALSE.	KINPP089

C	IF (ACTION .NE. NEW) GO TO 9	KINPP101
C	SET STANDARD OPTIONS	KINPP102
	4 CONC = .TRUE.	KINPP103
	EXCHR = .FALSE.	KINPP104
	COMBUS = .FALSE.	KINPP105
	SHOCK = .FALSE.	KINPP106
	TCON = .FALSE.	KINPP107
	RHOCON = .FALSE.	KINPP108
	ELIM = .FALSE.	KINPP109
	FVSTFP = .FALSE.	KINPP110
	DEBUG = .FALSE.	KINPP111
	EMAX = 0.0001	KINPP112
	ITPSZ = 5	KINPP113
	ALLMI = .FALSE.	KINPP114
		KINPP115
C	M(I,J) SET IN =BLKRDA	KINPP116
C		KINPP117
	BRIEF=.FALSE.	KINPP118
	TIMLMT=25.	KINPP119
		KINPP120
C		KINPP121
C	INITIALIZE	KINPP122
	NEXT = .FALSE.	KINPP123
	HINT = 0.	KINPP124
C	NLM SET IN =BLKEI	KINPP125
C	NS SET IN =BLKEI	KINPP126
C	LS SET IN =BLKRDA	KINPP127
C	LR SET IN =BLKRDA	KINPP128
	NT = 0	KINPP129
	DO 6 I=1,40	KINPP130
	XTB(I) = 0.	KINPP131
	CXTB(I) = 0.	KINPP132
	ATB(I) = 0.	KINPP133
	6 CATB(I) = 0.	KINPP134
	UNCEND = 0.	KINPP135
	CEND = 0.	KINPP136
C		KINPP137
C	STNIC(I,J) SET IN =BLKTH	KINPP138
C		KINPP139
	LSUBM = 0.	KINPP140
	ETA = 0.	KINPP141
	D = 0.	KINPP142
	VISC = 0.	KINPP143
	BETA = 0.	KINPP144
	CX3 = 0.	KINPP145
	CX2 = 0.	KINPP146
	CX1 = 0.	KINPP147
	CX0 = 0.	KINPP148
	GO TO 14	KINPP149
C		KINPP150

9 IF (ACTION,NE. CHANGE) GO TO 13	KINPP151
C REACTIONS AND REACTION RATE =	KINPP152
C REACTION RATE VARIABLES (A = N = FACT = LSR) =	KINPP153
C SET IN -RLKRDA-	KINPP154
C	KINPP155
13 IF (ACTION, EQ. REPEAT) GO TO 33	KINPP156
C	KINPP157
14 LSOLD = 0	KINPP158
LROLD = 0	KINPP159
C	KINPP160
C	KINPP161
C SPECIES NAME AND MOLECULAR WEIGHT SET IN -RLCK-	KINPP162
C	KINPP163
C	KINPP164
C	KINPP165
C	KINPP166
21 IF (ACTION, NE. NEW) GO TO 25	KINPP167
C INERT SPECIES SET IN -BLCK-	KINPP168
C	KINPP169
C SET INPUT SPECIES NAME EQUAL TO MASTER LIST NAME =	KINPP170
C SET COUNTERS (LS = LR = LSOLD) = SET WM AND YSS	KINPP171
C	KINPP172
LS = LSSV	KINPP173
LR = LRSV	KINPP174
LSOLD = LS	KINPP175
C	KINPP176
DO 23 II=1,LS	KINPP177
DSPNM(II) = DAISP(II)	KINPP178
MW(II) = ALMW(II)	KINPP179
YSS(II) = YI	KINPP180
23 CONTINUE	KINPP181
C	KINPP182
25 IF (LS, EQ. LSOLD) GO TO 30	KINPP183
C	KINPP184
C DETERMINE STOICHIOMETRIC COEFFICIENTS	KINPP185
C	KINPP186
C STOICHIOMETRIC COEFFICIENTS SET IN -RLKTH-	KINPP187
C	KINPP188
30 LRP = LROLD + 1	KINPP189
C	KINPP190
C GET SPECIES ENTHALPY AT REFERENCE T	KINPP191
TREF = 298.15	KINPP192
CALL THRM (TREF,0.)	KINPP193
C	KINPP194
TRAL = TREF*1.987165	KINPP195
DO 32 J=LRP,LR	KINPP196
N1 = LSR(1,J)	KINPP197
N2 = LSR(2,J)	KINPP198
N3 = LSR(3,J)	KINPP199
N4 = LSR(4,J)	KINPP200

C	COMPUTE HEAT OF REACTION	KINPP201
	DELH(J) = HRT(N3) - HRT(N2)	KINPP202
	IF (N1 .GT. 0) DELH(J) = DELH(J) + HRT(N1)	KINPP203
	IF (N4 .GT. 0) DELH(J) = DELH(J) + HRT(N4)	KINPP204
	32 DELH(J) = DELH(J)*TRAL	KINPP205
C		KINPP206
	LSP3 = L3 + 3	KINPP207
C	RESET STANDARD OPTIONS	KINPP208
	33 MOLEF = .TRUE.	KINPP209
	MMHG = .FALSE.	KINPP210
	NEWPR1 = .FALSE.	KINPP211
C		KINPP212
C	INITIALIZE	KINPP213
C	END = 0	KINPP214
	DELP = 0	KINPP215
	NTB = 0	KINPP216
	NPRINTS = 0	KINPP217
	DO 34 I=1,50	KINPP218
	34 APRINT(I) = 0	KINPP219
	AREA = 0	KINPP220
	MDOT = 0	KINPP221
C	P = 0	KINPP222
	V = 0	KINPP223
	RHD = 0	KINPP224
C	T = 0	KINPP225
	HN = 0	KINPP226
	FRRN = 0	KINPP227
	NH = 0	KINPP228
	AVH = 0	KINPP229
	JCV = 30	KINPP230
	KOUNT = 0	KINPP231
	DO 35 I=1,28	KINPP232
	RK(I) = 0	KINPP233
	35 F(I) = 0	KINPP234
C	I1 = 3PT IN -BLKRDA-	KINPP235
	I2 = 0	KINPP236
C		KINPP237
C	NAME OF INDEPENDENT VARIABLE = NAME OF ASSIGNED VARIABLE =	KINPP238
C	INPUT UNITS = OUTPUT UNITS = - SET IN NAMBLK	KINPP239
C		KINPP240
	IF (VERSA .EQ. BLANK) VERSA = AREAV	KINPP241
C		KINPP242
	IF (ACTION .NE. NEW) GO TO 80	KINPP243
C	INITIALIZE STEP SIZE LIMITS	KINPP244
	IF (VERSI .EQ. TIMEV) GO TO 78	KINPP245
	HMIN = 0.0001	KINPP246
	HMAX = 0.1000	KINPP247
	IPRCD = 2	KINPP248
	GO TO 79	KINPP249
	78 HMIN = 0.500E-07	KINPP250

HMAX = 0.500E+04	KINPP251
IPRCOD = 4	KINPP252
79 IF (VERSA .EQ. AREAV) IPRCOD = IPRCOD - 1	KINPP253
C READ INTEGRATION CONTROLS, PROFILE OPTIONS,	KINPP254
C PRINT OPTIONS, SPECIALTY SWITCHES	KINPP255
C 80 IF(FIRSTC) READ (5,PROB)	KINPP256
C	KINPP257
IF (.NOT. ALLMI) GO TO 36	KINPP258
DO 77 I=1,25	KINPP259
DO 77 J=1,30	KINPP260
77 M(I,J) = 1.	KINPP261
GO TO 40	KINPP262
C THIRD BODY RATIOS SET IN -BLKRDA=	KINPP263
36 CONTINUE	KINPP264
C	KINPP265
C SET INITIAL STEP SIZE	KINPP266
IF(FIRSTC) GO TO 40	KINPP267
HMIN = HMINZ	KINPP268
HMAX = HMAXZ	KINPP269
HINT = HINTZ	KINPP270
FMAX = FMAXZ	KINPP271
NPRINT = NPRNTZ	KINPP272
EVSTPZ = EVSTPZ	KINPP273
DBUGOZ = DBUGOZ	KINPP274
40 HI=HINT	KINPP275
IF(HINT.EQ.0.)HI=HMIN	KINPP276
C	KINPP277
C GET INITIAL CONDITIONS	KINPP278
C CALL INIT(TSS,MMHG,MOLF)	KINPP279
C IF(.NOT.FIRSTC) GO TO 41	KINPP280
HMINZ = HMIN	KINPP281
HMAXZ = HMAX	KINPP282
HINTZ = HINT	KINPP283
EMAXZ = EMAX	KINPP284
NPRNTZ = NPRINT	KINPP285
EVSTPZ = EVSTEP	KINPP286
DBUGOZ = DBUGO	KINPP287
CALL MOVE(1,PRINT,PRINTZ,50,1)	KINPP288
41 CONTINUE	KINPP289
DO 39 J=1,IR	KINPP290
39 R(J) = EACT(J)/1.987165	KINPP291
C	KINPP292
C SPFCES TO BE NEGLECTED FROM ERROR CONSIDERATIONS SET IN -BLKRDA=	KINPP293
C IT SET IN -BLKRDA=	KINPP294
C	KINPP295
C CHECK INPUT COMPOSITION	KINPP296
CSUM = 0.	KINPP297
DO 47 I=1,IS	KINPP298
47 CSUM = CSUM + C(I)	KINPP299
	KINPP300

IF (ABS(1.-CSUM) .LE. .001) GO TO 48	KINPP301
WRITE (6,105) CSUM,((SPNM(K,I),K=1,2),C(I),I=1,LS)	KINPP302
105 FORMAT (7H0(KINP),5H33HINVALID INPUT COMPOSITION SUM = ,F11.6//	KINPP303
* (12X,2A4,F20.5))	KINPP304
NEXT = .TRUE.	KINPP305
RETURN	KINPP306
C	KINPP307
48 IF (ITPSZ.GT.2)GO TO 53.	KINPP308
IF (ITPSZ.EQ.1 .AND. NTB.EQ.0) GO TO 53	KINPP309
IF (NTB.NE.0) NT = NTB	KINPP310
CONV = 1.	KINPP311
CON2 = 1.	KINPP312
IF (VERSA.NE. AREAIV) GO TO 203	KINPP313
XU = CU	KINPP314
AU(1) = CUA(1)	KINPP315
AU(2) = CUA(2)	KINPP316
C CONVERT AREA PROFILE TO INTERNAL UNITS	KINPP317
IF (UNIT1.NE. FPS) GO TO 201	KINPP318
XU = FU	KINPP319
AU(1) = FUA(1)	KINPP320
AU(2) = FUA(2)	KINPP321
CONV = 30.48	KINPP322
GO TO 202	KINPP323
201 IF (UNIT1.NE. SI) GO TO 206	KINPP324
XU = SU	KINPP325
AU(1) = SUA(1)	KINPP326
AU(2) = SUA(2)	KINPP327
CONV = 100.	KINPP328
202 CON2 = CONV*CONV	KINPP329
GO TO 206	KINPP330
C	KINPP331
203 XU = CU	KINPP332
AU(1) = CUP2(1)	KINPP333
AU(2) = CUP2(2)	KINPP334
C CONVERT PRESSURE PROFILE TO INTERNAL UNITS	KINPP335
IF (UNIT1.NE. FPS) GO TO 204	KINPP336
XU = FU	KINPP337
AU(1) = FUP(1)	KINPP338
AU(2) = FUP(2)	KINPP339
CONV = 30.48	KINPP340
CON2 = 1./2116.2	KINPP341
GO TO 205	KINPP342
204 IF (UNIT1.NE. SI) GO TO 205	KINPP343
XU = SU	KINPP344
AU(1) = SUP(1)	KINPP345
AU(2) = SUP(2)	KINPP346
CONV = 100.	KINPP347
CON2 = 1./1.01325E+05	KINPP348
205 IF (.NOT. MMHG) GO TO 206	KINPP349
AU(1) = CUP1(1)	KINPP350

AU(2) = CUP1(2)	KINPP351
CON2 = 1./760.	KINPP352
C	KINPP353
206 IF (VERST .EQ. TIMEV) CONV = 1.	KINPP354
IF (ITPS7 .EQ. 2) GO TO 208	KINPP355
DO 207 I=1,NTB	KINPP356
CXTB(I) = XTBI(I)*CONV	KINPP357
207 CATB(I) = ATB(I)*CON2	KINPP358
GO TO 53	KINPP359
208 DO 209 I=1,4	KINPP360
209 CN(I) = CX(I)*CON2	KINPP361
C	KINPP362
53 IF ((NPRNTS .NE. 0) .OR. (DELP .NE. 0)) .OR. (END .NE. 0)) NEWPR	KINPP363
* = .TRUE.	KINPP364
IF (.NOT. NEWPR) GO TO 59	KINPP365
IF (END .NE. 0) UNCEND = END	KINPP366
C PREPARE PRINT STATIONS	KINPP367
IF (FVSTEP) GO TO 59	KINPP368
IF (NPRNTS .NE. 0) GO TO 57	KINPP369
IF (DELP .NE. 0) GO TO 54	KINPP370
DELP = (UNCEND - IVAR)/24.9999	KINPP371
54 PRINT(I) = IVAR + DELP	KINPP372
DO 55 I=2,50	KINPP373
PRINT(I) = PRINT(I-1) + DELP	KINPP374
IF (PRINT(I) .GE. UNCEND) GO TO 56	KINPP375
55 CONTINUE	KINPP376
56 NP = I	KINPP377
PRINT(NP) = UNCEND	KINPP378
GO TO 59	KINPP379
C	KINPP380
57 NP = NPRNTS	KINPP381
IF (APRINT(I) .EQ. 0) GO TO 59	KINPP382
CONV = 1.	KINPP383
IF (TPRCND .EQ. 2 .OR. IPRCND .EQ. 4) GO TO 210	KINPP384
IF (UNIT1 .EQ. FPS) CONV = 30.48	KINPP385
IF (UNIT1 .EQ. ST) CONV = 100.	KINPP386
CON2 = CONV*CONV	KINPP387
GO TO 213	KINPP388
210 CON2 = 1.	KINPP389
IF (UNIT1 .NE. FPS) GO TO 211	KINPP390
CONV = 30.48	KINPP391
CON2 = 1./2116.2	KINPP392
GO TO 212	KINPP393
211 IF (UNIT1 .NE. ST) GO TO 212	KINPP394
CONV = 100.	KINPP395
CON2 = 1./1.01325E+05	KINPP396
212 IF (MMHG) CON2 = 1./760.	KINPP397
213 IF (VERST .EQ. TIMEV) CONV = 1.	KINPP398
CALL CURS (CATB,CXTB,NT).	KINPP399
DO 58 I=1,NPRNTS	KINPP400

	APRINT(I) = APRINT(I)*CON2	KINPP401
	CALL CIMP (APRINT(I),PRINT(I),DUM1,DUM2)	KINPP402
58	PRINT(I) = PRINT(I)/CONV	KINPP403
C		KINPP404
59	IF (ITPSZ .EQ. 1) CALL CUBS (CXTR,CATB,INT)	KINPP405
C		KINPP406
	IF (UNIT1 .NE. PPS) GO TO 63	KINPP407
C	CONVERT FROM FPS UNITS TO INTERNAL (CGS) UNITS	KINPP408
	IF (VERSI .NE. TIMEV) GO TO 60	KINPP409
	DVAR = DVAR*30.48	KINPP410
	GO TO 61	KINPP411
60	IVAR = IVAR*30.48	KINPP412
61	IF (MMHG) P = P*2.7845	KINPP413
	P = P/2116.2	KINPP414
	AREA = AREA*929.0304	KINPP415
	MDOT = MDOT*453.59237	KINPP416
	V = V*30.48	KINPP417
	RHO = RHO/62.43	KINPP418
	T = T/1.8	KINPP419
	IF ((.NOT. NEWPRY) .OR. VERSI .EQ. TIMEV) GO TO 68	KINPP420
	CEND = UNCFND*30.48	KINPP421
	DO 62 I=1,NP	KINPP422
62	PRINT(I) = PRINT(I)*30.48	KINPP423
	GO TO 68	KINPP424
63	IF (UNIT1 .NE. SI) GO TO 67	KINPP425
C	CONVERT FROM SI UNITS TO INTERNAL (CGS) UNITS	KINPP426
	IF (VERSI .NE. TIMEV) GO TO 64	KINPP427
	DVAR = DVAR*100.	KINPP428
	GO TO 65	KINPP429
64	IVAR = IVAR*100.	KINPP430
65	IF (MMHG) P = P*133.3224	KINPP431
	P = P/1.01325E+05	KINPP432
	AREA = AREA*10000.	KINPP433
	MDOT = MDOT*1000.	KINPP434
	V = V*100.	KINPP435
	RHO = RHO*.001	KINPP436
	IF ((.NOT. NEWPRY) .OR. VERSI .EQ. TIMEV) GO TO 68	KINPP437
	CEND = UNCFND*100.	KINPP438
	DO 66 I=1,NP	KINPP439
66	PRINT(I) = PRINT(I)*100.	KINPP440
	GO TO 68	KINPP441
C		KINPP442
67	CEND = UNCFND	KINPP443
	IF (MMHG) P = P/760.	KINPP444
C		KINPP445
68	MIXMH = 0.	KINPP446
C		KINPP447
	IF (.NOT. MOLEF) GO TO 71	KINPP448
C	MOLE FRACTION TO MOLES(I)/MASS(MIXTURF)	KINPP449
	DO 69 I=1,3	KINPP450

69	MIXMW = MIXMW + C(I)*MW(I)	KINPP451
	DO 70 I=1,13	KINPP452
70	SIGMA(I) = C(I)/MIXMW	KINPP453
	GO TO 73	KINPP454
C		KINPP455
C	MASS FRACTION TO MOLES(I)/MASS(MIXTURE)	KINPP456
71	DO 72 I=1,13	KINPP457
	SIGMA(I) = C(I)/MW(I)	KINPP458
72	MIXMW = MIXMW + SIGMA(I)	KINPP459
	MIXMW = 1./MIXMW	KINPP460
C		KINPP461
C	UNIVERSAL GAS CONSTANT IN ATM-CC/MOLE-DEG K	KINPP462
73	RR = 82.056	KINPP463
C	UNIVERSAL GAS CONSTANT IN ERGS/MOLE-DEG K	KINPP464
	R = 8.3143E+07	KINPP465
C		KINPP466
	IF (M2 .EQ. 0. .AND. .NOT. (COMBUS .OR. SHOCK)) GO TO 81	KINPP467
	CALL THRM (T,1.)	KINPP468
	CPR0 = 0.	KINPP469
	DO 74 I=1,13	KINPP470
74	CPR0 = CPR0 + CPR(I)*SIGMA(I)	KINPP471
	GAMMA = CPR0/(CPR0 + 1./MIXMW)	KINPP472
	IF (V .NE. 0.) GO TO 81	KINPP473
	V = SQRT(M2*R/MIXMW*GAMMA*T)	KINPP474
C		KINPP475
81	IF (P .EQ. 0.) GO TO 82	KINPP476
	RHO = P*MIXMW/(RR*T)	KINPP477
	GO TO 75	KINPP478
82	IF (RHO .EQ. 0.) GO TO 83	KINPP479
	P = RHO*RR*T/MIXMW	KINPP480
	GO TO 75	KINPP481
C		KINPP482
83	IF (IPRCD .GT. 2) GO TO 84	KINPP483
	X = TVAR	KINPP484
	IF (VERST .EQ. TIMEV) X = DVAR	KINPP485
	CALL C1NP (X,AVAR,DUM1,DUM2)	KINPP486
	GO TO 85	KINPP487
84	TIME = DVAR	KINPP488
	IF (VERST .EQ. TIMEV) TIME = TVAR	KINPP489
	CALL C1NP (TIME,AVAR,DUM1,DUM2)	KINPP490
85	IF (VERSA .EQ. AREAV) GO TO 86	KINPP491
	P = AVAR	KINPP492
	GO TO 81	KINPP493
86	AREA = AVAR	KINPP494
	RHO = MDO1/(AREA*V)	KINPP495
	GO TO 82	KINPP496
C		KINPP497
75	IF (MDO1 .EQ. 0.) MDO1 = RHO*AREA*V	KINPP498
C		KINPP499
C		KINPP500

RETURN
END

KINPP501
KINPP502

CLCFIT	INTEGRATE OR INTERPOLATE	'LCFIT'	LCFIT001
C			LCFIT002
C	INTEGRATE OR INTERPOLATE USING A PARABOLA WHICH PASSED THROUGH THE		LCFIT003
C	AND (I+1) POINTS BUT MISSES THE (I-1) AND (I+2) POINTS (IF THEY DO		LCFIT004
C	EXIST) SUCH THAT THE SQUARE OF THE DEVIATION IS A MINIMUM. NOTE		LCFIT005
C	THAT I IS GENERALLY SELECTED SUCH THAT		LCFIT006
C	X(I), LE, XC, LT, X(I+1)		LCFIT007
C	THE EQUATION FOR THE PARABOLA IS		LCFIT008
C	$Y-Y(I) = B*(X-X(I)) + C*(X-X(I))^2$		LCFIT009
C			LCFIT010
	SUBROUTINE LCFIT(X,Y,NPTS, NEW, XC,YC,NXC,ND, C)		LCFIT011
	DIMENSION X(10),Y(10), XC(10),YC(10), C(10)		LCFIT012
	LOGICAL NEW		LCFIT013
C			LCFIT014
C	INPUT-		LCFIT015
C	X, Y PTS. ON CURVE		LCFIT016
C	NPTS NO. OF X		LCFIT017
C	NEW = .FALSE. IF THE 'C' ARRAY OF COEFFICIENTS IS		LCFIT018
C	AVAILABLE FROM A PREVIOUS ENTRY		LCFIT019
C	XC LIST OF X AT WHICH CALC TO BE DONE		LCFIT020
C	YC(1) INTEGRATION CONSTANT IF NDC=1		LCFIT021
C	NXC NO. OF XC		LCFIT022
C	ND =0 TO GET COORD, =1 TO GET 1ST DERIVATIVE,		LCFIT023
C	=-1 FOR INTEGRATION		LCFIT024
C			LCFIT025
C	OUTPUT		LCFIT026
C	YC COORDINATE OR DERIVATIVE AT XC OR		LCFIT027
C	YC(IC)= INTEGRAL(Y*DX) FROM XC(1) TO XC(IC) WHERE IC=2,NXC		LCFIT028
C	C = ARRAY OF (NPTS-1) COEFFICIENTS		LCFIT029
C			LCFIT030
C	NOTES-		LCFIT031
C	FOR INTEGRATION 'XC' MUST BE IN THE SAME ORDER AS 'X'. FOR INTERP		LCFIT032
C	'X' MAY BE IN EITHER ASCENDING OR DESCENDING ORDER.		LCFIT033
C	NO SPECIAL ORDER IS REQUIRED		LCFIT034
C			LCFIT035
C	COMMON /CLSPF / I		LCFIT036
C			LCFIT037
	LOGICAL WITHIN		LCFIT038
	DATA BITS/0377777777777777/		LCFIT039
	IF(.NOT.NEW) GO TO 90		LCFIT040
	N = NPTS-1		LCFIT041
C			LCFIT042
C	EVALUATE COEFFICIENT C(I)		LCFIT043
C			LCFIT044
	DO 30 I=1,N		LCFIT045
	XI = X(I)		LCFIT046
	YI = Y(I)		LCFIT047
	X3 = X(I+1)-XI		LCFIT048
	BUT = 0.		LCFIT049
	TDP = 0.		LCFIT050

C(I) = 0.	LCFIT051
Y3 = Y(I+1)-YI	LCFIT052
IF(I.LE.1 .OR. Y(I-1).EQ.BITS) GO TO 27	LCFIT053
X1 = X(I-1)-XI	LCFIT054
X13 = X(I-1)-X(I+1)	LCFIT055
TOP = X1*(Y3*X1-(Y(I-1)-YI)*X3)*X13	LCFIT056
BOT = X1*X1*X13*X13*X3	LCFIT057
27 IF(I.GE.N .OR. Y(I+2).EQ.BITS) GO TO 28	LCFIT058
271 X4 = X(I+2)-XI	LCFIT059
X43 = X(I+2)-X(I+1)	LCFIT060
TOP = TOP + X4*(Y3*X4-(Y(I+2)-YI)*X3)*X43	LCFIT061
BOT = BOT + X4*X4*X43*X43*X3	LCFIT062
28 IF(BOT.NE.0.) C(I)=-TOP/BOT	LCFIT063
30 CONTINUE	LCFIT064
C	LCFIT065
C BEGIN INTERPOLATION LOOP FOR XC(IC) IC=1,NXC	LCFIT066
90 I = MAX(1,MJNO(I,N))	LCFIT067
IF(ND.EQ.(-1)) I=1	LCFIT068
SGN = SIGN(1.,X(N+1)-X(1))	LCFIT069
IC = 1	LCFIT070
GO TO 160	LCFIT071
C	LCFIT072
C LOCATE APPROPRIATE INTERVAL	LCFIT073
100 WITHIN= .FALSE.	LCFIT074
NCOUNT= N	LCFIT075
102 IF(NCOUNT) 119,103,103	LCFIT076
103 NCOUNT= NCOUNT-1	LCFIT077
C	LCFIT078
XI = X(I)	LCFIT079
XD = XC(IC)-XI	LCFIT080
IF(N) 104,120,104	LCFIT081
104 IF(SGN*XD) 105,107,110	LCFIT082
C	LCFIT083
C F.IY.0. (F IS THE FRACTIONAL POSITION IN THE INTERVAL)	LCFIT084
105 IF(I.EQ.1) GO TO 120	LCFIT085
IF(ND.EQ.(-1)) GO TO 119	LCFIT086
I = I-1	LCFIT087
GO TO 102	LCFIT088
C	LCFIT089
C F.EQ.0.	LCFIT090
107 IF(X(I+1).NE.XI) GO TO 120	LCFIT091
GO TO 114	LCFIT092
C	LCFIT093
C F.GT.0.	LCFIT094
110 IF(SGN*(XC(IC)-X(I+1))) 120,112,114	LCFIT095
C	LCFIT096
C F.EQ.1.0, CHECK FOR INTEGRATION AND DOUBLE POINT BEFORE INCREMENT	LCFIT097
112 IF((ND.EQ.(-1)) .OR. (I.NE.N .AND. X(I+1).EQ.X(I+2))) GO TO 120	LCFIT098
C	LCFIT099
C F.GT.1.0	LCFIT100
114 IF(I.EQ.N) GO TO 120.	

IF(ND.EQ.(-1)) GO TO 125	LCFIT101
116 I = I+1	LCFIT102
GO TO 102	LCFIT103
C	LCFIT104
119 CALL ERROR1	LCFIT105
C	LCFIT106
C PRELIMINARY CALCULATIONS FOR INTERPOLATION OR INTEGRATION	LCFIT107
120 WITHIN=TRUE.	LCFIT108
125 X3 = X(I+1)-X(I)	LCFIT109
YI = Y(I)	LCFIT110
B = 0.	LCFIT111
C1 = C(I)	LCFIT112
IF(N.GT.0 .AND. X3.NE.0.) B = (Y(I+1)-YI)/X3 = C1*X3	LCFIT113
129 IF(ND) 130,140,141	LCFIT114
C	LCFIT115
C ND=-1, INTEGRATE	LCFIT116
130 IF(.NOT.WITHIN) XD=X3	LCFIT117
S1 = (YI + (B/2. + C1/3.*XD)*XD)*XD	LCFIT118
IF(WITHIN) GO TO 135	LCFIT119
C 'I' IS BEING INCREMENTED TO FIND APPROPRIATE INTERVAL. HENCE,	LCFIT120
C CUMULATE THE INTEGRAL OF THE ITH INTERVAL.	LCFIT121
SA = SA + S1	LCFIT122
GO TO 116	LCFIT123
C APPROPRIATE INTERVAL FOUND. X(I)-XC(IC)=X(I+1)	LCFIT124
135 IF(IC.EQ.1) SA=YC(IC)=S1	LCFIT125
IF(IC.NE.1) YC(IC)=SA+S1	LCFIT126
GO TO 150	LCFIT127
C	LCFIT128
C ND=0, INTERPOLATE FOR COORDINATES	LCFIT129
140 YC(IC)= YI + (B + C1*XD)*XD	LCFIT130
GO TO 150	LCFIT131
C	LCFIT132
C ND=1, FIRST DERIVATIVE	LCFIT133
141 YC(IC)= B + 2.*C1*XD	LCFIT134
GO TO 150	LCFIT135
C	LCFIT136
150 IC = IC+1	LCFIT137
160 IF(NXC-IC) 900,100,100	LCFIT138
C	LCFIT139
900 RETURN	LCFIT140
END	LCFIT141

CLESVV	GEN'L DBL PREC LINEAR EQN SOLVER	LESVV001
	SUBROUTINE LESV (K)	LESVV002
C		LESVV003
C	THIS ROUTINE IS A GENERAL DOUBLE PRECISION LINEAR EQUATION SOLVER	LESVV004
C	IN THIS PROGRAM IT IS USED TO COMPUTE THE INCREMENTS K(I)	LESVV005
C		LESVV006
C	DOUBLE PRECISION A,S,TS,B	LESVV007
C		LESVV008
C	LOGICAL NEXT	LESVV009
C		LESVV010
C	REAL K	LESVV011
C		LESVV012
C	DIMENSION S(28),K(28)	LESVV013
C		LESVV014
C	COMMON/COND/DUMMY(34),N,NEXT	LESVV015
C	COMMON/MATX/A(28,29)	LESVV016
C		LESVV017
C	NP = N+1	LESVV018
C	DO 5 I=1,N	LESVV019
C	GET SCALE FACTORS	LESVV020
C	TS = 0.00	LESVV021
C	DO 6 J=1,N	LESVV022
C	B = DABS(A(I,J))	LESVV023
C	IF (B .GT. TS) TS = B	LESVV024
C	6 CONTINUE	LESVV025
C	IF (TS .EQ. 0.00) GO TO 100	LESVV026
C		LESVV027
C	SCALE ROWS	LESVV028
C	DO 10 J=1,NP	LESVV029
C	10 A(I,J) = A(I,J)/TS	LESVV030
C	5 CONTINUE	LESVV031
C		LESVV032
C	BEGIN TRIANGULARIZATION	LESVV033
C	IF (N .EQ. 1) GO TO 25	LESVV034
C	NM = N-1	LESVV035
C	DO 15 J=1,NM	LESVV036
C	FIND MAXIMUM ELEMENT IN COLUMN J BELOW DIAGONAL	LESVV037
C	II = J	LESVV038
C	JP = J+1	LESVV039
C	DO 16 I=JP,N	LESVV040
C	IF (DABS(A(I,J)) .GT. DABS(A(II,J))) II = I	LESVV041
C	16 CONTINUE	LESVV042
C	IF (II .EQ. J) GO TO 20	LESVV043
C	INTERCHANGE ROWS II AND J	LESVV044
C	DO 17 L=J,NP	LESVV045
C	TS = A(II,L)	LESVV046
C	A(II,L) = A(J,L)	LESVV047
C	17 A(J,L) = TS	LESVV048
C		LESVV049
C	ZERO COLUMN J BELOW DIAGONAL	LESVV050

20 DO 18 J=JP,N	LESVV051
TS = A(I,J)/A(J,J)	LESVV052
IF (TS .EQ. 0.00) GO TO 18	LESVV053
DO 19 L=JP,NP	LESVV054
19 A(I,L) = A(I,L) - TS*A(J,L)	LESVV055
18 CONTINUE	LESVV056
15 CONTINUE	LESVV057
C	LESVV058
C BACK SUBSTITUTE	LESVV059
25 TS = A(N,N)	LESVV060
IF (DABS(TS) .LT. 1.D-10) WRITE (6,102)	LESVV061
102 FORMAT (7H0(LESV),5X,11HSINGULARITY)	LESVV062
S(N) = A(N,NP)/TS	LESVV063
K(N) = S(N)	LESVV064
IF (N .EQ. 1) RETURN	LESVV065
DO 26 I=2,N	LESVV066
M = NP - I	LESVV067
TS = A(M,M)	LESVV068
IF (DABS(TS) .LT. 1.D-10) WRITE (6,102)	LESVV069
B = A(M,NP)	LESVV070
MP = M+1	LESVV071
DO 27 L=MP,N	LESVV072
27 B = B - A(M,L)*S(L)	LESVV073
S(M) = B/TS	LESVV074
26 K(M) = S(M)	LESVV075
30 CONTINUE	LESVV076
RETURN	LESVV077
C	LESVV078
100 WRITE (6,101) I	LESVV079
101 FORMAT (7H0(LESV),3X,3HROW,14,39H OF THE COEFFICIENT MATRIX IS ALL	LESVV080
1 ZERO3)	LESVV081
DO 50 J=1,N	LESVV082
50 WRITE (6,103) (A(I,J), J=1,NP)	LESVV083
103 FORMAT (1H1,8E16.6/(1X,8E16.6))	LESVV084
NEXT = .TRUE.	LESVV085
RETURN	LESVV086
C	LESVV087
END	LESVV088

*LSPFIT DUMMY--CALLS LFIT
SUBROUTINE LSPFIT (X,Y,N,XC,YC,NP,I)
CALL LFIT (X,Y,N,XC,YC,NP,I)
RETURN
END

LSPFIT01
LSPFIT02
LSPFIT03
LSPFIT04
LSPFIT05

CLFIT		LFIT0001
	SUBROUTINE LFIT(X,Y,NPTS,XC,YC,NXC,ND)	LFIT0002
CLFIT	INTEGRATE OR INTERPOLATE	LFIT0003
*	INTEGRATE OR INTERPOLATE USING A STRAIGHT LINE FIT BETWEEN POINTS	LFIT0004
C		LFIT0005
C		LFIT0006
C*	CHECK TO KEEP -LFIT- FROM EXTRAPOLATION--	LFIT0007
C*	IF REQUESTED INTERPOLATION LOCATION IS OUTSIDE DEFINED INTERVAL, SET	LFIT0008
C*	VARIABLE EQUAL TO END VALUE	LFIT0009
C		LFIT0010
C		LFIT0011
C		LFIT0012
	DIMENSION X(10),Y(10), XC(10),YC(10)	LFIT0013
C		LFIT0014
C	INPUT-	LFIT0015
C	X, Y PTS. ON CURVE	LFIT0016
C	NPTS NO. OF X	LFIT0017
C	XC LIST OF X AT WHICH CALC TO BE DONE	LFIT0018
C	YC(1) INTEGRATION CONSTANT IF ND=-1	LFIT0019
C	NXC NO. OF XC	LFIT0020
C	ND =0 TO GET COORD, =1 TO GET 1ST DERIVATIVE,	LFIT0021
C	=-1 FOR INTEGRATION	LFIT0022
C		LFIT0023
C	OUTPUT	LFIT0024
C	YC COORDINATE OR DERIVATIVE AT XC OR	LFIT0025
C	YC(IC)= INTEGRAL(Y*DX) FROM XC(1) TO XC(IC) WHERE IC=2,NX	LFIT0026
C		LFIT0027
C	NOTES	LFIT0028
C	'X' MAY BE IN EITHER ASCENDING OR DESCENDING ORDER.	LFIT0029
C	FOR INTEGRATION 'XC' MUST BE IN THE SAME ORDER AS 'X'.	LFIT0030
C	FOR INTERPOLATION NO SPECIAL ORDER IS REQUIRED.	LFIT0031
C		LFIT0032
	COMMON /CLSPF / I	LFIT0033
	LOGICAL WITHIN	LFIT0034
C		LFIT0035
	N = NPTS-1	LFIT0036
	I = MAX(1,MIN(1,N))	LFIT0037
	IF (ND.EQ.(-1)) I=1	LFIT0038
	ISAVE = 0	LFIT0039
	SGN = SIGN(1.,X(N+1)-X(1))	LFIT0040
C		LFIT0041
C	BEGIN INTERPOLATION LOOP FOR XC(IC) IC=1,NXC	LFIT0042
	IC = 1	LFIT0043
C		LFIT0044
C	LOCATE APPROPRIATE INTERVAL	LFIT0045
	100 WITHIN= .FALSE.	LFIT0046
	NCOUNT= N	LFIT0047
	102 IF (NCOUNT) 119,103,103	LFIT0048
	103 NCOUNT= NCOUNT-1	LFIT0049
	XI = X(I)	LFIT0050

XD = XC(IC)-XI	LFIT0051
IF (N) 104,120,104	LFIT0052
104 IF (SGN*XD)105,107,110	LFIT0053
C	LFIT0054
F.LT.0. (F IS THE FRACTIONAL POSITION IN THE INTERVAL)	LFIT0055
C	LFIT0056
C	LFIT0057
105 IF (I.EQ.1) GO TO 117	LFIT0058
C*105 IF (I.EQ.1) GO TO 120	LFIT0059
C	LFIT0060
C	LFIT0061
IF (ND.EQ.(-1)) GO TO 119	LFIT0062
I = I-1	LFIT0063
GO TO 102	LFIT0064
C	LFIT0065
C F.EQ.0.	LFIT0066
107 IF (X(I+1).NE.XI) GO TO 120	LFIT0067
GO TO 116	LFIT0068
C F.GT.0.	LFIT0069
110 IF (SGN*(XC(IC)-X(I+1))) 120,112,114	LFIT0070
C	LFIT0071
C F.EQ.1., CHECK FOR INTEGRATION AND DOUBLE POINT BEFORE INCREMENTIN	LFIT0072
112 IF ((ND.EQ.(-1)) .OR. (I.NE.N .AND. X(I+1).EQ.X(I+2))) GO TO 120	LFIT0073
C	LFIT0074
C F.GT.1.	LFIT0075
C	LFIT0076
C	LFIT0077
114 IF (I.EQ.N) GO TO 118	LFIT0078
C*114 IF (I.EQ.N) GO TO 120	LFIT0079
C	LFIT0080
C	LFIT0081
IF (ND.EQ.(-1)) GO TO 122	LFIT0082
116 I = I+1	LFIT0083
GO TO 102	LFIT0084
C	LFIT0085
C	LFIT0086
C	LFIT0087
C* EXTRAPOLATION OUTSIDE X-TABLE---USE END VALUES	LFIT0088
117 YJ=Y(1)	LFIT0089
GO TO 1181	LFIT0090
C	LFIT0091
118 YI=Y(N+1)	LFIT0092
1181 B=0.	LFIT0093
C	LFIT0094
C	LFIT0095
GO TO 129	LFIT0096
C	LFIT0097
C	LFIT0098
C	LFIT0099
119 CALL ERRDRI	LFIT0100

*LFIT1	LINEAR FIT INTERPOLATION	*LFIT10	LFIT1001
	SUBROUTINE LFIT1(X,Y,NPTS, XC,YC,NXC)		LFIT1002
	DIMENSION X(10),Y(10), XC(10),YC(10)		LFIT1003
C			LFIT1004
C	INPUT-		LFIT1005
C	X,Y = LIST OF COORDINATES DESCRIBING THE INPUT FUNCTION		LFIT1006
C	NPTS = NUMBER OF X,Y POINTS		LFIT1007
C	XC = LIST OF X-S AT WHICH INTERPOLATION IS TO BE PERFORMED		LFIT1008
C	NXC = NUMBER OF XC-VALUES		LFIT1009
C			LFIT1010
C	OUTPUT-		LFIT1011
C	YC = LIST OF VALUES INTERPOLATED AT XC(IC),IC=1,NXC		LFIT1012
C			LFIT1013
C	NOTES-		LFIT1014
C	IF XC IS OUTSIDE OF THE RANGE OF X, THE END VALUE OF Y IS USED FOR YC.		LFIT1015
C	X MUST BE LISTED FROM SMALLEST TO LARGEST.		LFIT1016
C	DOUBLE X-POINTS ARE ALLOWED FOR A FUNCTION DISCONTINUITY.		LFIT1017
C			LFIT1018
C	N = NPTS		LFIT1019
C	I = 1		LFIT1020
C			LFIT1021
C	BEGIN INTERPOLATION LOOP FOR XC(IC),IC=1,NXC		LFIT1022
C	IC = 1		LFIT1023
	60 XCIC = XC(IC)		LFIT1024
	IF(N.GT.1) GO TO 100		LFIT1025
	YC(IC)=Y(1)		LFIT1026
	GO TO 190		LFIT1027
C			LFIT1028
	100 XG = X(I+1)-XCIC		LFIT1029
	IF(XG) 114,114,102		LFIT1030
	102 XF = XCIC-X(I)		LFIT1031
	IF(XF) 110,120,120		LFIT1032
C			LFIT1033
C	F,LT,0. (F IS THE FRACTIONAL POSITION IN THE INTERVAL)		LFIT1034
	110 I = I-1		LFIT1035
	IF(I) 100,111,100		LFIT1036
	111 I = 1		LFIT1037
	YC(IC)=Y(1)		LFIT1038
	GO TO 190		LFIT1039
C			LFIT1040
C	F,GE,1.		LFIT1041
	114 I = I+1		LFIT1042
	IF(I-N) 100,115,100		LFIT1043
	115 I = N+1		LFIT1044
	YC(IC)=Y(N)		LFIT1045
	GO TO 190		LFIT1046
C			LFIT1047
C	INTERPOLATE		LFIT1048
	120 YC(IC)= (Y(I)*XG+Y(I+1)*XF)/(XG+XF)		LFIT1049
			LFIT1050

C
C INDEX TO NEXT XC(IC)
190 IC = IC+1
IF(IC.LE.NXC) GO TO 60

C
RETURN
END

LFIT1051
LFIT1052
LFIT1053
LFIT1054
LFIT1055
LFIT1056
LFIT1057

C		LFIT0101
C	PRELIMINARY CALCULATIONS FOR INTERPOLATION OR INTEGRATION	LFIT0102
	120 WITHIN= .TRUE.	LFIT0103
	122 IF (1-1SAVE) 124,129,124	LFIT0104
	124 1SAVE = 1	LFIT0105
	YI = Y(I)	LFIT0106
	X3 = X(I+1)-XI	LFIT0107
	B = 0.	LFIT0108
	IF (N.GT.0 .AND. X3.NE.0.) B=(Y(I+1)-YI)/X3	LFIT0109
	129 IF (ND) 130,140,141	LFIT0110
C		LFIT0111
C	ND=-1, INTEGRATE	LFIT0112
	130 IF (.NOT.WITHIN) XD=X3	LFIT0113
	S1 = (YI+.5*B*XD)*XD	LFIT0114
	IF (WITHIN) GO TO 135	LFIT0115
C	'I' IS BEING INCREMENTED TO FIND APPROPRIATE INTERVAL. HENCE,	LFIT0116
C	CUMULATE THE INTEGRAL OF THE ITH INTERVAL.	LFIT0117
	SA = SA+S1	LFIT0118
	GO TO 116	LFIT0119
C	APPROPRIATE INTERVAL FOUND. X(I)-XC(IC)-X(I+1)	LFIT0120
	135 IF (IC.EQ.1) SA=YC(IC)-S1	LFIT0121
	IF (IC.NE.1) YC(IC)=SA+S1	LFIT0122
	GO TO 150	LFIT0123
C	ND=0, INTERPOLATE FOR COORDINATES	LFIT0124
	140 YC(IC) = YI+B*XD	LFIT0125
	GO TO 150	LFIT0126
C	ND=1, FIRST DERIVATIVE	LFIT0127
	141 YC(IC) = B	LFIT0128
C		LFIT0129
	150 IC = IC+1	LFIT0130
	IF (NXC-IC) 900,160,160	LFIT0131
	160 IF (ND.NE.(-1) .AND. XC(IC).EQ.XC(IC-1)) I=I+1	LFIT0132
	GO TO 100	LFIT0133
C		LFIT0134
	900 RETURN	LFIT0135
	END	LFIT0136

CHAINBD MAIN BLOCK DATA
BLOCK DATA
COMMON /TROUBL/ DERR(q)
LOGICAL DERR
COMMON /CBITS / BITS,BLANK

E

DATA DERR/4*F/
DATA BITS,BLANK/03777777777,6H /
END

MAINBD01
MAINBD02
MAINBD03
MAINBD04
MAINBD05
MAINBD06
MAINBD07
MAINBD08
MAINBD09

CHAINJ	MAIN JETMIX SUBROUTINE	MAINJ001
	SUBROUTINE MAIN	MAINJ002
	COMMON /ADAM01/ NAM(40)	MAINJ003
	COMMON /TAG / NAM1(50)	MAINJ004
	COMMON /CNERR / BITS1,KERR,DUMV(3)	MAINJ005
	LOGICAL KERR	MAINJ006
	COMMON /CBITS / BITS,BLANK	MAINJ007
	COMMON /TROUBL/ ERR,ERRMAJ,INERR,PRERR	MAINJ008
	LOGICAL FRR,ERRMAJ,INERR,PRERR	MAINJ009
	COMMON /CARRY / NEW	MAINJ010
	LOGICAL NEW	MAINJ011
	COMMON /FILK / CSC	MAINJ012
	COMMON /DIFENI/ DIF(122)	MAINJ013
	COMMON /BCMJL / BCM(24)	MAINJ014
C		MAINJ015
	BITS1 = BITS	MAINJ016
	NEW = .TRUE.	MAINJ017
	CSC = 1000.	MAINJ018
	KERR = .FALSE.	MAINJ019
	CALL SFTM(1,BLANK,NAM1(21),30)	MAINJ020
	CALL MOVE(1,NAM,NAM1,20,1)	MAINJ021
C		MAINJ022
	CALL LLINK(6HINLINK)	MAINJ023
	CALL JTINIT	MAINJ024
C		MAINJ025
C	READ NAMELIST \$A	MAINJ026
C		MAINJ027
	1 CALL JETINP	MAINJ028
	IF(.NOT,KERR) GO TO 2	MAINJ029
	ERR = .TRUE.	MAINJ030
	RETURN	MAINJ031
C		MAINJ032
	2 CALL LLINK(6HCPLINK)	MAINJ033
	CALL JTCIRL	MAINJ034
	IF(KERR) ERR=.TRUE.	MAINJ035
	GO TO 3	MAINJ036
C		MAINJ037
	ENTRY ERRORI	MAINJ038
	FRR = .TRUE.	MAINJ039
	3 CALL JTDUTS	MAINJ040
C		MAINJ041
	RETURN	MAINJ042
	END	MAINJ043

CMAINP MAIN PROGRAM--EMIS DRIVER

LOGICAL EOF	MAINP001
COMMON /IDFILE/ TAPIN,TAPOT	MAINP002
LOGICAL TAPIN,TAPOT	MAINP003
COMMON /CPRGDM/ PRGDM	MAINP004
INTEGER PRGDM	MAINP005
COMMON /TROUBL/ ERR,ERRMAJ,INERR,PRERR	MAINP006
LOGICAL ERR,ERRMAJ,INERR,PRERR	MAINP007
COMMON /ADAM02/ ENDJOB,DUMMY(3)	MAINP008
LOGICAL ENDJOB	MAINP009
COMMON /ADAM01/ NAME(10),ADDRES(10),TITLE(10),IDENT(10)	MAINP010
COMMON /CNFILE/ NF	MAINP011
INTEGER PREJET,GMOD,SPALDG,EMIS,CFILE	MAINP012
DATA PREJET,JETMIX,GMOD,SPALDG,EMIS,CFILE/	MAINP013
* 6HPREJET,6HJETMIX,4HGMOD,6HSPALDG,4HEMIS,5HCFILE/	MAINP014
	MAINP015
C	MAINP016
C CALL FLGEDF(5,EOF)	MAINP017
C	MAINP018
C READ ADAM01 BLOCK	MAINP019
C	MAINP020
5 READ (5,20) NAME	MAINP021
READ (5,20) ADDRES	MAINP022
READ (5,20) IDENT	MAINP023
20 FORMAT(1X,10A6)	MAINP024
22 TAPOT = .FALSE.	MAINP025
TAPIN = .FALSE.	MAINP026
READ (5,25) PRGDM,TAPIN,TAPOT	MAINP027
IF(EOF) GO TO 1000	MAINP028
25 FORMAT(1X,A6,4X,L1,1X,L1)	MAINP029
30 WRITE (6,35) PRGDM,TAPIN,TAPOT	MAINP030
35 FORMAT(1H1,10X,16HEXECUTING PRGDM=A6/	MAINP031
* 10X,6HTAPIN=L2,5X,6HTAPOT=L2/)	MAINP032
	MAINP033
C	MAINP034
C SELECT PROGRAM	MAINP035
C	MAINP036
40 IGO = 1	MAINP037
IF(PRGDM,EQ,JETMIX) IGO=2	MAINP038
IF(PRGDM,EQ,GMOD) IGO=3	MAINP039
IF(PRGDM,EQ,SPALDG) IGO=4	MAINP040
IF(PRGDM,EQ,EMIS) IGO=5	MAINP041
IF(PRGDM,EQ,CFILE) IGO=6	MAINP042
50 GO TO (100,200,250,300,400,500) , IGO	MAINP043
C	MAINP044
C PREJET*****	MAINP045
C	MAINP046
100 IF(TAPOT) REWIND 1	MAINP047
101 CALL LLINK(PREJET)	MAINP048
CALL MAIN	MAINP049
IF(TAPOT) ENDFILE 1	MAINP050

IF(ERR) GO TO 600	MAINP051
GO TO 22	MAINP052
C JETMIX*****	MAINP053
C	MAINP054
C	MAINP055
200 IF((.NOT.TAPIN),AND,(.NOT,TAPDT)) GO TO 201	MAINP056
REWIND 1	MAINP057
REWIND 2	MAINP058
REWIND 3	MAINP059
201 CALL LLINK(JETMIX)	MAINP060
CALL MAIN	MAINP061
IF(TAPDT) ENDFILE 2	MAINP062
IF(ERR) GO TO 600	MAINP063
GO TO 22	MAINP064
C	MAINP065
C GMD*****	MAINP066
C	MAINP067
250 REWIND 1	MAINP068
REWIND 3	MAINP069
CALL LLINK(4HGMJD)	MAINP070
CALL MAIN	MAINP071
ENDFILE 1	MAINP072
GO TO 22	MAINP073
C	MAINP074
C SPALDG*****	MAINP075
C	MAINP076
300 IF(TAPIN) REWIND 2	MAINP077
IF(TAPIN) REWIND 1	MAINP078
IF(TAPDT) REWIND 3	MAINP079
301 CALL LLINK(SPALDG)	MAINP080
CALL MAIN	MAINP081
IF(TAPDT) ENDFILE 3	MAINP082
IF(ERR) GO TO 600	MAINP083
GO TO 22	MAINP084
C	MAINP085
C EMIS*****	MAINP086
C	MAINP087
400 REWIND 1	MAINP088
REWIND 2	MAINP089
REWIND 3	MAINP090
401 CALL LLINK(EMIS)	MAINP091
CALL MAIN	MAINP092
IF(ERR) GO TO 600	MAINP093
GO TO 22	MAINP094
C	MAINP095
C CFILE*****	MAINP096
C	MAINP097
500 REWIND 1	MAINP098
REWIND 2	MAINP099
REWIND 3	MAINP100

IF(TAPOT) REWIND 4	MAINP101
IF(TAPIN) REWIND 7	MAINP102
501 CALL LLINK(CFILE)	MAINP103
CALL MAIN	MAINP104
IF(ERR, AND, PROGM, EQ, CFILE) GO TO 600	MAINP105
REWIND 1	MAINP106
REWIND 2	MAINP107
REWIND 3	MAINP108
IF(ERR, AND, (PROGM, EQ, JETHIX, OR, PROGM, EQ, SPALOG)) GO TO 1002	MAINP109
GO TO 22	MAINP110
C	MAINP111
C	MAINP112
C	MAINP113
600 WRITE(6, 601) PROGM	MAINP114
601 FORMAT(// 2X, 9HERR = T, 5X, 6HPRGMS, A6//	MAINP115
* 16X, 21H** RUN TERMINATION **)	MAINP116
IF(TAPOT, AND, (PROGM, EQ, JETHIX, OR, PROGM, EQ, SPALOG)) GO TO 500	MAINP117
1000 WRITE(6, 1001)	MAINP118
1001 FORMAT(// 10X, 26H***** ENDJOB *****)	MAINP119
1002 STOP	MAINP120
END	MAINP121

CMFMMAIN	-MFMAIN- MAIN SUB. FOR MXFLUT	MFMAIN01
C		MFMAIN02
C		MFMAIN03
C	NOV. 1973	MFMAIN04
C	SUBROUTINE MFMAIN	MFMAIN05
C		MFMAIN06
C		MFMAIN07
C****	COMMON /CINPUT/ XX(5),R(5,50),U(5,50),RHQ(5,50),XSPECI(5,50,7),	MFMAIN08
	COMMON /CINPUT/ XY(5),R(5,50),U(5,50),RHQ(5,50),XSPECI(5,50,13),	MFMAIN09
	1PPSI(5,50),GG(5,50),NJ(5),NII	MFMAIN10
	COMMON /CSPECI/ NSPECI,NF,DX	MFMAIN11
	COMMON /JETDAT/ NTUBES,YREACT(12),TS(12),UREACT(12),SPV(12),	MFMAIN12
	1ZHWI(12),CP(12),FSPECI(12),GHAT(12),TKE(12),OTHER(36)	MFMAIN13
	COMMON /GASCOMP/ GAS(1104)	MFMAIN14
	COMMON /CAXIAL/ X,XL,ALOGX	MFMAIN15
	COMMON /CXIACA/ XX(5)	MFMAIN16
	COMMON /CPCTRL/ TITLE(20),PPRINT(30)	MFMAIN17
	COMMON /CINPJT/ DIAJ,DUMIX(114)	MFMAIN18
	COMMON /STCTRL/ LSTA,FINAL,CHEMK,FIRSTM,DUMST1(3),NIST,POUT1,	MFMAIN19
	* ALFLIM,CMIXST,DUMST2(6)	MFMAIN20
	LOGICAL FIRSTM,FINAL	MFMAIN21
	COMMON /PORIDE/ PSTA(30)	MFMAIN22
	COMMON /CBITS/ BITS,BLANK	MFMAIN23
	COMMON /CDXSAV/ DXSAVE	MFMAIN24
	COMMON /XISAVE/ IXXX,ISTARX	MFMAIN25
	COMMON /CMXFLT/ XFRAC(12,12),XFRHAT(13,12),ALPHA(12,12)	MFMAIN26
	COMMON /CPRINT/ PDUM(20)	MFMAIN27
	DATA NSX/0/	MFMAIN28
C		MFMAIN29
	NAMELIST /INPUT/ X,XL,NTUBES,DX,ALFLIM,CMIXST,NIST	MFMAIN30
	IF (.NOT. FIRSTM) GO TO 250	MFMAIN31
	X=0.	MFMAIN32
	NQ=0	MFMAIN33
	XL=1.2	MFMAIN34
	DX=.05	MFMAIN35
	NIST = 10	MFMAIN36
	READ (5,INPUT)	MFMAIN37
	DXSAVE=DX	MFMAIN38
	NII=5	MFMAIN39
	CMIXST= 2.	MFMAIN40
100	CALL READI	MFMAIN41
	IF(PSTA(1).EQ.BITS) GO TO 200	MFMAIN42
	DO 101 J=1,30	MFMAIN43
	IF(PSTA(J).EQ.BITS) GO TO 200	MFMAIN44
	PPRINT(J)= PSTA(J)*DIAJ/12.	MFMAIN45
101	CONTINUE	MFMAIN46
200	ALOGX=ALOG(X+1.)	MFMAIN47
	CALL NEWNET	MFMAIN48
	IF(PDUM(19).NE.0. .AND. X.GE.PDUM(19)) PDUM(17)=1.	MFMAIN49
	CALL MXFLUT	MFMAIN50

	IF(PDUM(18).NE.0.) CALL MPRINT	MFMAIN51
	IF(FIRSTM) GO TO 250	MFMAIN52
C		MFMAIN53
C	ADJUST STEP SIZE	MFMAIN54
C		MFMAIN55
	ALFLIM= .95	MFMAIN56
	NTUBF1= NTUBES-1	MFMAIN57
	ALFMIN= 2.	MFMAIN58
	DO 240 J=1,NTUBE1	MFMAIN59
	ALFMIN= AMIN,(ALFMIN,ALPHA(J,J))	MFMAIN60
240	CONTINUE	MFMAIN61
	IF(ALFMIN.GT.ALFLIM) DXSAVE= CMIXSTADXSAVE	MFMAIN62
250	XL= PPRINT(LSIA)*12./DIAJ	MFMAIN63
	FIRSTM= .FALSE.	MFMAIN64
	IF (X,GE,XL) GO TO 310	MFMAIN65
	NSX = NSX+1	MFMAIN66
	IF(NSX.GT.NIST) GO TO 311	MFMAIN67
	DX=DXSAVE	MFMAIN68
	X=X+DX	MFMAIN69
	IF (X.LE.XL) GO TO 260	MFMAIN70
	DX=XL-(X-DX)	MFMAIN71
	X=XL	MFMAIN72
260	IF (X.LT.XX(NII-2)) GO TO 200	MFMAIN73
	IF(NII.LE.4 .AND. X.LT.XX(NII)) GO TO 200	MFMAIN74
C*	MOVE OLD DATA BACK IN STORAGE AND READ IN NEW DATA	MFMAIN75
	DO 300 I=1,4	MFMAIN76
	XX(I)=XX(I+1)	MFMAIN77
	XY(I)=XY(I+1)	MFMAIN78
	DO 299 J=1,50	MFMAIN79
	R(I,J)=R(I+1,J)	MFMAIN80
	U(I,J)=U(I+1,J)	MFMAIN81
	RHO(I,J)=RHO(I+1,J)	MFMAIN82
	DO 298 K=1,NF	MFMAIN83
298	XSPECI(I,J,K)=XSPECI(I+1,J,K)	MFMAIN84
	GG(I,J)=GG(I+1,J)	MFMAIN85
299	PPSI(I,J)=PPSI(I+1,J)	MFMAIN86
300	NJJ(I)=NJJ(I+1)	MFMAIN87
	GO TO 100	MFMAIN88
C		MFMAIN89
C*	--FINAL=T-- INDICATED WE HAVE REACHED A PRINT STATION	MFMAIN90
310	FINAL = .TRUE.	MFMAIN91
	RETURN	MFMAIN92
311	XDIFF = XL-X	MFMAIN93
	IF(XDIFF.LE.6.E-7) GO TO 310	MFMAIN94
	RETURN	MFMAIN95
	END	MFMAIN96

CMSHCUT	MESH REFINEMENT ROUTINE	MSHCUT01
	SUBROUTINE MSHCUT(NREG,YM,NPD)	MSHCUT02
	LOGICAL MCHANG	MSHCUT03
	COMMON /PRDF/	MSHCUT04
	* DIM(400),UD(200),THD(200),ED(200)	MSHCUT05
	COMMON /UMFSH/ MCHANG,CK,DY1,NM3H,	MSHCUT06
	COMMON /MISC/ PM(10)	MSHCUT07
		MSHCUT08
C*****C*		MSHCUT09
C*	COMMON /CTRL2/ DUM24(2), MERGE , DUM26(6)	MSHCUT10
	COMMON /RCO/ UD, EO, THO	MSHCUT11
	COMMON /JET2/ TWD, DUM55(7)	MSHCUT12
	LOGICAL TWD, MERGE	MSHCUT13
C*		MSHCUT14
	COMMON /MIXER/ MIX,RO(100),XD(100),CF,YR(100)	MSHCUT15
	LOGICAL MIX	MSHCUT16
	COMMON /GLOBAL/ MAXIT,SUPB,NIT,PSID,YDD,YDC,	MSHCUT17
	* P1,P2,UCL,TOL,UPSTRM,CVG	MSHCUT18
	LOGICAL SUPB,CVG,UPSTRM	MSHCUT19
	COMMON /CNERR/ BITS,ERR,GC,GCJ,FOOT	MSHCUT20
	LOGICAL ERR	MSHCUT21
	COMMON /ACONVG/ YCD(100),PD(100),INDC(100), CHOKE, CHOKED	MSHCUT22
	LOGICAL CHOKE, CHOKED	MSHCUT23
	COMMON /DFIT/ CLSP(100)	MSHCUT24
	COMMON /STA2/ MACH2,TS2,SS2,V2,RHO2,OPDX2	MSHCUT25
	REAL MACH2	MSHCUT26
	COMMON /RCMIX2/ GRADU,TW,MUW,RHOW,PTF,TTE	MSHCUT27
	REAL MUW	MSHCUT28
	COMMON /CBODY/ YCB(100),CLSPCB(100),YCB1 , UCL1	MSHCUT29
C*		MSHCUT30
C*****C*		MSHCUT31
	DIMENSION IMP(10)	MSHCUT32
	EQUIVALENCE (IMP(1),PM(1))	MSHCUT33
	DIMENSION YM(1)	MSHCUT34
C*		MSHCUT35
C*	REDUCE NO. OF POINTS IN DOWNSTREAM MESH BY NRED	MSHCUT36
C*		MSHCUT37
	IF(IMP(1),F0.0) GO TO 1	MSHCUT38
	CALL TABPRT(4HPSIB,YM,NPD,10,0)	MSHCUT39
	CALL TABPRT(2HUD,UD,NPD,10,0)	MSHCUT40
	CALL TABPRT(3HTHD,THD,NPD,10,0)	MSHCUT41
	CALL TABPRT(2HED,ED,NPD,10,0)	MSHCUT42
	1 KGD=NREG	MSHCUT43
	10 YE=YM(NPD)	MSHCUT44
	IF(.NOT. MCHANG) GO TO 11	MSHCUT45
	GO TO (11,20,20), KGD	MSHCUT46
C*		MSHCUT47
C*	POTENTIAL CORE REGION--	MSHCUT48
C*	RECALCULATE NEW UNIFORM MESH.	MSHCUT49
C*		MSHCUT50

11 LK=2	MSHCUT51
IF(.NOT. TWO) GO TO 6312	MSHCUT52
TEST1=U0-1,E=6	MSHCUT53
TEST2=U0+1,E=6	MSHCUT54
DO 6311 L=1,NPD	MSHCUT55
IF(UD(L).GE.TEST1 .AND. UD(L).LE.TEST2) UD(L)=U0	MSHCUT56
6311 CONTINUE	MSHCUT57
6312 IF(.NOT. MCHANG) GO TO 166	MSHCUT58
DO 14 L=2,NPD	MSHCUT59
IF(UD(L).NE. UCL1) GO TO 16	MSHCUT60
14 CONTINUE	MSHCUT61
16 LK=L	MSHCUT62
C*	MSHCUT63
IF((.NOT. TWO) .OR. MERGE) GO TO 166	MSHCUT64
DO 17 L=LK,NPD	MSHCUT65
IF(UD(L).EQ.U0) GO TO 18	MSHCUT66
17 CONTINUE	MSHCUT67
18 LK=L	MSHCUT68
DO 19 L=LK,NPD	MSHCUT69
IF(UD(L).NE. U0) GO TO 167	MSHCUT70
19 CONTINUE	MSHCUT71
167 LK=L	MSHCUT72
166 LK1=LK-1	MSHCUT73
NPD=NPD-NRED	MSHCUT74
DY=(YE-YM(LK1))/FLOAT(NPD-LK1)	MSHCUT75
DO 15 L=LK,NPD	MSHCUT76
15 YM(L)=YM(L-1)+DY	MSHCUT77
GO TO 100	MSHCUT78
C*	MSHCUT79
C* TRANSITION/SIMILAR REGION-- CALCULATE NEW DY=1	MSHCUT80
C*	MSHCUT81
20 NPD=NPD-NRED	MSHCUT82
STEPS=FLOAT(NPD-1)	MSHCUT83
DYC=YE/(CK**STEPS-1.)*(CK-1.)	MSHCUT84
DO 30 L=2,NPD	MSHCUT85
XXP=FLOAT(L-1)	MSHCUT86
30 YM(L)=DYC*(CK**XXP-1.)/(CK-1.)	MSHCUT87
C*	MSHCUT88
100 IF(IMP(1).EQ.0) GO TO 101	MSHCUT89
CALL TABPRT(4HPSIA,YM,NPD,10.0)	MSHCUT90
101 RETURN	MSHCUT91
END	MSHCUT92

CMXFLTI	INITIATE HETEROGENEOUS GAS MODEL AT EACH PROFILE SURVEY	MXFLTI01
C	POINT IN ENGINE EXHAUST PLANE.	MXFLTI02
C	SUBROUTINE MXFLTO(J)	MXFLTI03
	DIMENSION ZI(11),SMWT(16),ZL(16),ZR(11), QV1(8),QV2(8)	MXFLTI04
	REAL N,KF,MACHR,MACHL,MWTJ	MXFLTI05
	INTEGER TLGO	MXFLTI06
	LOGICAL ALFUF	MXFLTI07
	COMMON/JETDAT/NPTJ,RAJ(12),TSJ(12),IJ(12),SPVJ(12),MWTJ(12),	MXFLTI08
	CPJ(12),FUELJ(12),SPLNGJ(12),TKEJ(12),OTHRJ(12,3)	MXFLTI09
	COMMON/GASCMP/YYK(12,2),FK(2,12,2),HK(2,12,2),ZK(16,2,12,2),	MXFLTI10
	HCTNCP(12,2),OTHRK(2,12,2,4)	MXFLTI11
	COMMON/COTREM/YIOL,YO,DYDX,CTRMAY/DOITREM/NDQT	MXFLTI12
	COMMON/INDATA/N,HF,KAR,T2,HETA,T25,FARS,EINDC,P0,	MXFLTI13
	RCD(11),RCD2(11),RHC(11),RNOX(11),PT(11),PS(11),BLNC(11),FICDC(11)	MXFLTI14
	DATA SMWT/ 1.008,16.000, 2.016,32.000,17.008,18.016,28.010,44.010,	MXFLTI15
	28.016,59.944,30.008,14.008,33.008,46.008,17.032,00.000/	MXFLTI16
	DATA ZR,FICD,EINDC,H2,SL,WTL1,H25,SR,WTR,CPEF,DH,HFV,CPEV/25*0.0/	MXFLTI17
	DATA AFR,SPVR/2*0.0/	MXFLTI18
	AVG(A,B)=YY*A+(1.0-YY)*B	MXFLTI19
	FGAM(G)=SQRT(G*(2.0/(G+1.0))*((G+1.0)/(G-1.0)))	MXFLTI20
C		MXFLTI21
C	START THE FLIGHT RECORDER	MXFLTI22
C		MXFLTI23
	NDQT=40	MXFLTI24
	REWINO 40	MXFLTI25
C		MXFLTI26
C	REDUCE GAS ANALYSIS DATA	MXFLTI27
C		MXFLTI28
	CALL CAROL(N,WAR,RHC(J),RCD(J),RCD2(J),RNOX(J),FBARS,EIHC,FICD,	MXFLTI29
	EINDC)	MXFLTI30
	RH2CD=RH2(RCD(J))/RCD(J)	MXFLTI31
	SMWT(16)=120.1+10.08*N	MXFLTI32
	BLNC1=FARS/FHARS/(1.0-FICD*0.1*SMWT(16)/28.01-EIHC)-1.0	MXFLTI33
	BLNC(J)=AMAX1(BLNC(J),BLNC1)	MXFLTI34
	FHARS=FBARS/(1.0+FBARS+WAR)	MXFLTI35
	FMAX=0.00146457*SMWT(16)	MXFLTI36
	FMAX=FMAX/(1.0+FMAX+WAR)	MXFLTI37
C		MXFLTI38
C	LOCAL WET AIR ENTHALPY (BLNC IS LOCAL BY-PASS RATIO)	MXFLTI39
C		MXFLTI40
	CALL EOGAST(0.0,WAR,N,T2,PS(J),.FALSE.,.FALSE.,Z1,H2,	MXFLTI41
	WTL1,SI,SPVL,AFL,CPEI)	MXFLTI42
	CALL EOGAST(0.0,WAR,N,T25,PS(J),.FALSE.,.FALSE.,Z1,H25,	MXFLTI43
	WTL1,SI,SPVL,AFL,CPEI)	MXFLTI44
	HAS=(1.0+HETA)*H2-BETA*H25	MXFLTI45
	HATR=(HAS+BLNC(J)*H25)/(1.0+BLNC(J))	MXFLTI46
C		MXFLTI47
C	INITIATED AIR COMPOSITION	MXFLTI48
C		MXFLTI49
		MXFLTI50

F5=FARS/(1.0+BLOC(J))	MXFLT151
CALL SFTM(1,0,0,Z1,11)	MXFLT152
Z1(11)=F5*FINOPC/46.008	MXFLT153
Z1(7)=F5*E1COC(J)/28.01	MXFLT154
Z1(3)=R42C0*Z1(7)	MXFLT155
Z1(10)=0.00032189	MXFLT156
Z1(9)=0.02695794-0.5*Z1(11)	MXFLT157
Z1(5)=5.0*N*F5/SMWT(16)+WAR/18.016-Z1(3)	MXFLT158
Z1(8)=10.0*F5/SMWT(16)+1.036E-5-Z1(7)	MXFLT159
Z1(4)=0.00724264+WAR/36.032-Z1(8)-0.5*(Z1(6)+Z1(7)+Z1(11))	MXFLT160
CALL FMPYC(1,1.0/(1.0+F5+WAR),Z1,Z1,11)	MXFLT161
WTAIR=0.0	MXFLT162
DO 10 IS=1,11	MXFLT163
10 WTAIR=WTAIR+Z1(IS)	MXFLT164
WTAIR=1.0/WTAIR	MXFLT165
C	MXFLT166
C MAXIMUM CO THAT MODEL CAN CRFATE	MXFLT167
C	MXFLT168
FG5=F5/(1.0+F5+WAR)	MXFLT169
A=(FBARS-FG5)/(1.0-FG5)	MXFLT170
E1COMX=(A/FBARS-FINC)*10.0/SMWT(16)*28.01	MXFLT171
+E1COC(J)*FG5*(1.0-A)/FBARS	MXFLT172
IF(E1COC(J).E1COMX)GO TO 30	MXFLT173
WRITE(6,9996)J,E1COC,E1COMX	MXFLT174
9996 FORMAT(15H0MXFLT0 - PT NO,13,27H, CANNOT MODEL MEASURED CO ,	MXFLT175
143HWITHUT DISSOCIATING TURBINE DISCHARGE GAS,10X,	MXFLT176
117HETON REDUCED FROM,1PE10.3,3H TO,E10.3)	MXFLT177
E1COC=E1COMX	MXFLT178
30 CONTINUE	MXFLT179
C	MXFLT180
C MAXIMUM MASS FRACTION RICH GAS IN SAMPLE	MXFLT181
C	MXFLT182
YYMAX=1.0-FBARS*F1NC	MXFLT183
C	MXFLT184
C YY AT WHICH RICH GAS IS STOICHIOMETRIC	MXFLT185
C	MXFLT186
FARS=0.1*SMWT(16)/28.966643*0.209495/(1.0+0.25*N)	MXFLT187
FSTOIC=FARS/(1.0+FARS+WAR)	MXFLT188
YYSTC=(FBARS-FG5-(1.0-FG5)*FBARS*F1NC)/(FSTOIC-FG5)	MXFLT189
YYSTEP=0.1	MXFLT190
IF(FBARS.LT.FSTOIC)YYSTEP=YYMAX-YYSTC	MXFLT191
C	MXFLT192
C INITIALIZE BFFORF ITERATING	MXFLT193
C	MXFLT194
YYS=YYMAX	MXFLT195
U=0.0	MXFLT196
TL=1800.0	MXFLT197
TR=3000.0	MXFLT198
QV2(1)=0.0	MXFLT199
C	MXFLT200

C	COMMENCE ITERATION FOR RICH FRACTION	MXFLT101
C		MXFLT102
40	IF(YYS.GT.YYMAX) YYS=YYMAX	MXFLT103
	A=FBARS*FIHC	MXFLT104
	YYMIN=(FBARS-FG5-A*(1.0-FG5))/(FMAX-FG5)	MXFLT105
	IF(YYS.LT.YYMIN) YYS=YYMIN	MXFLT106
	FUNLIT=1.0	MXFLT107
	IF(YYS.LT.YYMAX) FUNLIT=A/(1.0-YYS)	MXFLT108
	FL=FUNLIT*FG5*(1.0-FUNLIT)	MXFLT109
	FR=(FBARS-(1.0-YYS)*FL)/YYS	MXFLT110
C		MXFLT111
	HTL=(1.0-FL)*HATR+FL*HF	MXFLT112
	HTR=(1.0-FR)*HATR+FR*HF	MXFLT113
	FARR=FR*(1.0+WAR)/(1.0-FR)	MXFLT114
	IF(FARR.LT.0.0) CALL ERROR1	MXFLT115
	CALL EQGASH(FARR,WAR,N,1TR,PT(J),,FALSE,,FALSE,,ZR,	MXFLT116
	HTR,HTR,SR,SPVR,AFR,CPR)	MXFLT117
C		MXFLT118
C	LEAN GAS COMPOSITION	MXFLT119
C		MXFLT120
	CALL SFTM(1,0.0,ZL,16)	MXFLT121
	CALL EMPYC(1,1.0-FUNLIT,71,ZL,11)	MXFLT122
	ZL(16)=FUNLIT/SMWT(16)	MXFLT123
	WTL=0.0	MXFLT124
	DO 60 IS=1,16	MXFLT125
60	WTL=WTL+ZL(IS)	MXFLT126
	WTL=1.0/WTL	MXFLT127
	HL=HTL	MXFLT128
	ASSIGN 50 TO TLGO	MXFLT129
	GO TO 80	MXFLT130
50	TTRQL=TTR/TL	MXFLT131
C		MXFLT132
C	COMMENCE ITERATION FOR VELOCITY	MXFLT133
C		MXFLT134
	QV1(1)=0.0	MXFLT135
70	KE=U*2/50072.884	MXFLT136
C		MXFLT137
C	RICH GAS PROPERTIES	MXFLT138
C		MXFLT139
	HR=HTR-TTRQL*KE/(1.0+(TTRQL-1.0)*YYS)	MXFLT140
	ZR(11)=(FBARS*FINOX/46.008-(1.0-YYS)*ZL(11))/YYS	MXFLT141
	IF((ZR(11).LT.0.0).OR.(YYS.EQ.YYMIN)) ZR(11)=0.0	MXFLT142
	ZR(9)=0.02695/94*(1.0-FR)/(1.0+WAR)-0.5*ZR(11)	MXFLT143
	CALL EQGASH(FARR,WAR,N,1TR,PS(J),,FALSE,,TRUE,,ZR,	MXFLT144
	HR,HTR,SR,SPVR,AFR,CPR)	MXFLT145
	C=CPR*HTR/1.98596	MXFLT146
	GAMR=C/(C-1.0)	MXFLT147
	MACHR=U/AFR	MXFLT148
	TTR=1R*KE/CPR	MXFLT149
	PTBPS=PRAT/MACHR,GAMR)	MXFLT150

C		MXFLT151
C	LEAN GAS PROPERTIES	MXFLT152
C	HL=HTL-KE/(1.0+(TTRQL-1.0)*YYS)	MXFLT153
	ASSIGN 115 TO TLGD.	MXFLT154
80	ALFUEL=FUNLIT.EQ.1.0	MXFLT155
	DO 100 I(CNT=1,30	MXFLT156
	IF(TL,IT,189.0)TL=189.0	MXFLT157
	CALL FRGAST(TL,PS(J),Z1,HLX,WT1,SI,SPVL,AFL,CPFL)	MXFLT158
	CALL FUEL(N,TL,HFV,CPFV)	MXFLT159
	A=1.0-FUNLIT	MXFLT160
	HLX=FUNLIT*HFV+A*HLX	MXFLT161
	CPFL=FUNLIT*CPFV+A*CPFL	MXFLT162
	SPVL=Z1(.6)*10.73191*TL/PS(J)+A*SPVL	MXFLT163
	DT=(HL-HLX)/CPFL	MXFLT164
	IF(ABS(DT).LT.0.05)GO TO 110	MXFLT165
	IF((TL.EQ.189.0).AND.(DT.LT.0.0))GO TO 105	MXFLT166
	IF(ABS(DT).GT.500.0)DI=SIGN(500.0,DT)	MXFLT167
100	TL=TL+DT	MXFLT168
105	IF(ALFUEL)GO TO 110	MXFLT169
	WRITE(6,9999)J,DT,FL,TL,HL,YYS,U	MXFLT170
9999	FORMAT(51H04XFLT10 - LEAN GAS TEMP ITERATION DID NOT CONVERGE./	MXFLT171
	16H0PT NO,13,6H DT=,1PE12,4,5H FL=,E12,4,5H TL=,E12,4,	MXFLT172
	15H HL=,F12,4,6H YYS=,E12,4,4H U=,F12,4)	MXFLT173
110	GO TO TLGD.(50,115)	MXFLT174
115	C=CPFL*WT1/1.98596	MXFLT175
	GAML=C/(C-1.0)	MXFLT176
	AFL=SQRT(49/21.372*GAML/WT1*TL)	MXFLT177
	MACH1=U/AFL	MXFLT178
	TTL=TL+KE/CPFL	MXFLT179
	PTLPS=PRAT(MACH1,GAML)	MXFLT180
C		MXFLT181
C	FRFE STREAM MASS AND VOLUME FRACTIONS OF RICH GAS	MXFLT182
C	A=FGAM(GAMR)/FGAM(GAML)*PTRPS/PTLPS*SQRT(WTR/WT1*TTL/TTR)	MXFLT183
	B=SPVR/SPVL	MXFLT184
	YY=1.0+A*B*(1.0-YYS)/YYS	MXFLT185
	YY=1.0/YY	MXFLT186
	TAU=B*YY/(1.0+YY*(B-1.0))	MXFLT187
C		MXFLT188
C	CALCULATED APPARENT IMPACT PRESSURE	MXFLT189
C	PTPSX=(PTRPS*TAU**2+PTLPS*(1.0-TAU)**2)/(TAU**2+(1.0-TAU)**2)	MXFLT190
	WRITE(40,9998)J,YYS,EICQX,U,PTPSX,EICJC(J),FL,HTL,TL	MXFLT191
	MACH1,PTLPS,ZL,FR,HTR,TR,MACHR,PTRPS,ZR	MXFLT192
9998	FORMAT(132X/10H ***** J=,13,6H YYS=,F7,4,8H EICQX=,1PE10,3,	MXFLT193
	14H U=,0PFR,1,8H PTPSX=,1PE10,3,	MXFLT194
	16H QCD=,E10,3,8H ***** 34X/20X,3HFL=,	MXFLT195
	E10,3,6H HTL=,0PFR,1,5H TL=,F8,1,8H MACHL=,1PE10,3,8H PTPSR=,	MXFLT196
	E10,3,7H XLR=,29X/2X,13F10,3,72X,3F10,3,100X/20X,3HFR=,E10,3,	MXFLT197
		MXFLT198
		MXFLT199
		MXFLT200

16H	HTR=,0PF8,1,5H	IR=,FR,1,8H	MACHR=,1PF10,3,8H	PTRPS=,E10,3,	MXFLT101
17H	XR=,29X/2X,11E10,3,20X)				MXFLT102
	Y0=PT(J)/PS(J)				MXFLT103
	YTOL=1E-4*Y0				MXFLT104
	DYDX=1.0/AVG(AFR,AFL)				MXFLT105
	CALL QTRFM(U,PTPSX,-2000.,QV1)				MXFLT106
	IF(QV1(1).NE.0.0)GO TO 70				MXFLT107
C					MXFLT108
C	CHECK THE CALCULATED SAMPLE CO CONTENT				MXFLT109
C					MXFLT110
	EICDX=(YYS*ZR(7)+(1.0-YYS)*ZL(7))*28.01/FBARS				MXFLT111
	IF((YYS.FN.YYMAX).AND.(EICDX.GT.EICO))GO TO 120				MXFLT112
	Y0=ALOG(EICO)				MXFLT113
	Y=-50.0				MXFLT114
	IF(EICDX.GT.0.0)Y=ALOG(EICDX)				MXFLT115
	YTOL=1E-3				MXFLT116
	DYDX=0.0				MXFLT117
	CALL QTRFM(YYS,Y,YYSTEP,QV2)				MXFLT118
	IF(QV2(1).NE.0.0)GO TO 40				MXFLT119
C					MXFLT120
C	AVERAGE POINT PROPERTIES FOR JETMIX				MXFLT121
C					MXFLT122
	120 FUELJ(J)=AVG(FR,FL)				MXFLT123
	SPLOGJ(J)=AVG((FR-FUELJ(J))*2,(FL-FUELJ(J))*2)				MXFLT124
	CPJ(J)=AVG(CPFR,CPFL)				MXFLT125
	MWTJ(J)=1.0/AVG(1.0/MTR,1.0/MTL)				MXFLT126
	SPVJ(J)=AVG(SPVR,SPVL)				MXFLT127
C	144/1,98596/778,16=0.09318008				MXFLT128
	TSJ(J)=0.09318008*PS(J)*SPVJ(J)*MWTJ(J)				MXFLT129
	UJ(J)=U				MXFLT130
C					MXFLT131
C	DETAILED 2-PART HETEROGENEOUS GAS PROPERTIES FOR MXFLUT				MXFLT132
C					MXFLT133
	YYK(J,1)=YY				MXFLT134
	FK(1,J,1)=FR				MXFLT135
	FK(2,J,1)=FL				MXFLT136
	HK(1,J,1)=HR				MXFLT137
	HK(2,J,1)=HL				MXFLT138
	MCINCP(J,1)=0.0				MXFLT139
	CALL MOVE(2,ZL,ZK(1,2,J,1),16,1,ZR,ZK(1,1,J,1),11,1)				MXFLT140
	CALL SFTM(1,0.0,7K(12,1,J,1),5)				MXFLT141
C					MXFLT142
	RETURN				MXFLT143
	END				MXFLT144

*MXFLUT		MXFLUT01
SUBROUTINE MXFLUT		MXFLUT02
C		MXFLUT03
LOGICAL NETRO		MXFLUT04
COMMON /CMATRX/ INTR1(12),INTR2 ,DET,IFACTR		MXFLUT05
COMMON /CRFACT/ RHOREA(12)		MXFLUT06
COMMON /CMXFLT/ XFRACT(12,12),XFRHAT(13,12),ALPHA(12,12)		MXFLUT07
COMMON /CSPECI/ NSPECI,NF,DX		MXFLUT08
COMMON /CYOLD/ YRICH(12)		MXFLUT09
COMMON /CNWNET/ FDUM(100),FUELFL(12),FDUMM(100)		MXFLUT10
COMMON /CCECHK/ SPECIF(12),SPECIS(13),RATIN(12)		MXFLUT11
C		MXFLUT12
COMMON /JETDAT/ NTURES,YREACT(12),TS(12),UREACT(12),SPV(12),		MXFLUT13
*ZMWT(12),CP(12),FGSPEC(12),GHAT(12),TKE(12),TFR(12),TFL,		MXFLUT14
*TIM1(12),TIM2,OTHER1(10)		MXFLUT15
COMMON /GASCOMP/ YRICH(12),YPRIME(12),FOAIR(2,12,2),ENTH(2,12,2),		MXFLUT16
*CONC(16,2,12,2),HCINCP(12,2), U(12,2),DUMSG2(168)		MXFLUT17
COMMON /GASTHW/ DUMTM1(96),TAU(12,2),DUMTM2(48)		MXFLUT18
COMMON /CMASS/ ZMASS(12),ZMASSM(12),FUELL(12),FUELR(12)		MXFLUT19
COMMON /CPRINT/ PDUM(20)		MXFLUT20
COMMON /SICTRL/ I STA,DUMST(16)		MXFLUT21
COMMON /CAXIAL/ XCC,DUMAX(2)		MXFLUT22
COMMON /CAGAIN/ AGAIN,IENTRY		MXFLUT23
DIMENSION ERROR(12)		MXFLUT24
DIMENSION CONC1(768)		MXFLUT25
EQUIVALENC (CONC1(1),CONC(1,1,1,1))		MXFLUT26
C		MXFLUT27
DATA NCNC /16/		MXFLUT28
DATA JOIM,JOID,LNEW,JRICH,JLEAN/12,1,2,1,2/		MXFLUT29
C		MXFLUT30
DO 11 IC=1,768		MXFLUT31
IF(CONC1(IC).LT.0.) CONC1(IC)=0.		MXFLUT32
11 CONTINUE		MXFLUT33
NRFACT=NTURES		MXFLUT34
IENTRY=IENTRY+1		MXFLUT35
C		MXFLUT36
IF (IENTRY.GT.1) GO TO 100		MXFLUT37
CALL MOVE(1,UREACT,U(1,LOLD),12,1)		MXFLUT38
100 CONTINUE		MXFLUT39
C		MXFLUT40
C* --FGSPEC-- IS THE FUEL/GAS RATIO OF SPECIE *K*		MXFLUT41
C		MXFLUT42
C* ADD ENTRAINE MASS FLOW TO TOTAL MASS OF AMBIENT GAS SPECIES		MXFLUT43
SPECIS(VSPECI)=SPECIS(NSPECI)+XFRACT(NSPECI,NSPECI)		MXFLUT44
C		MXFLUT45
C* SCALE FLOW OF EACH SPECT TO INSURE EXACT CONSERVATION		MXFLUT46
DO 210 K=1,NSPECI		MXFLUT47
SPECIF(K)=0.		MXFLUT48
C* SUM FLOW OF SPECIES *K* OVER ALL TUBES		MXFLUT49
DO 200 J=1,NTURES		MXFLUT50

200 SPECIF(K)=SPECIF(K)+XFRHAT(K,J)	MXFLUT51
C	MXFLUT52
210 CONTINUE	MXFLUT53
IF(PDUM(16).NE.0.) CALL TABPRT(6H\$SPECIF,SPECIF,NF,10)	MXFLUT54
C* SAVE ORIGINAL AMOUNT OF FLOW OF SPECIES(K) IN **SPECIS(K)	MXFLUT55
C	MXFLUT56
IF(PDUM(17).EQ.0.) GO TO 2100	MXFLUT57
WRITE (6,2)	MXFLUT58
WRITE (6,6)	MXFLUT59
WRITE (6,4)	MXFLUT60
WRITE (6,5) (K,K=1,12)	MXFLUT61
WRITE (6,1) (J,(XFRAC(T(K,J),K=1,12),J=1,NSPECI)	MXFLUT62
2100 CONTINUE	MXFLUT63
C	MXFLUT64
CALL ALFA2(NTUBES)	MXFLUT65
C	MXFLUT66
IF(PDUM(17).EQ.0.) GO TO 2200	MXFLUT67
WRITE (6,2)	MXFLUT68
WRITE (6,8)	MXFLUT69
WRITE (6,9) (K,K=1,12)	MXFLUT70
WRITE (6,3) (J,(ALPHA(K,J),K=1,12),J=1,NSPECI)	MXFLUT71
C	MXFLUT72
WRITE (6,2)	MXFLUT73
WRITE (6,7)	MXFLUT74
WRITE (6,4)	MXFLUT75
WRITE (6,5) (K,K=1,12)	MXFLUT76
WRITE (6,1) (J,(XFRHAT(K,J),K=1,12),J=1,NSPECI)	MXFLUT77
2200 CONTINUE	MXFLUT78
C	MXFLUT79
C	MXFLUT80
C* **ZMASS** IS MASS FLOW IN TUBE -J-	MXFLUT81
C* **FUELI** IS MASS OF FUEL IN TUBE -J- IN LEAN CLASS	MXFLUT82
C* **FUELR** IS MASS OF FUEL IN TUBE -J- IN RICH CLASS	MXFLUT83
C* **FLEAN** IS MIXTURE RATIO IN LEAN CLASS	MXFLUT84
C* **FRICH** IS MIXTURE RATIO IN RICH CLASS	MXFLUT85
C	MXFLUT86
C	MXFLUT87
C	MXFLUT88
C* --FOAIR-- IS FUEL TO GAS RATIO	MXFLUT89
C	MXFLUT90
C* SET MASS FLOW IN DUMMY OUTER TUBE	MXFLUT91
ZMASS(NSPECI)=XFRAC(T(NSPECI,NSPECI)	MXFLUT92
C* RECONSTRUCT MASS FLOW IN EACH TUBE AT NEW STATION	MXFLUT93
DO 220 J=1,NTUBES	MXFLUT94
ZMASSH(J)= 0.	MXFLUT95
DO 220 K=1,NSPECI	MXFLUT96
220 ZMASSH(J)= ZMASSH(J)+ZMASS(K)*ALPHA(K,J)	MXFLUT97
CALL MOVE (1,UREACT,U(1,LNEW),12,1)	MXFLUT98
DO 411 J=1,NTUBES	MXFLUT99
FUELI=0.	MXFLUT00

FUELR=0.
ZMASSL=0.
ZMASSR=0.

MXFLUT01
MXFLUT02
MXFLUT03

C

DO 301 L=1, NSPECI

MXFLUT04
MXFLUT05

C

C* PERFORM UST MIXING STEP

MXFLUT06
MXFLUT07

C* CALCULATE MASS OF FUEL IN EACH TUBE AND EACH CLASS

MXFLUT08

FUFLI =ZMASS(L)*ALPHA(L,J)*FOAIR(JIFAN,L,LOLD)*(1.-YRICH(L))+

MXFLUT09

IFUELL

MXFLUT10

FUFLR =ZMASS(L)*ALPHA(L,J)*FOAIR(JRICH,L,LOLD)*YRICH(L)+FUFLR

MXFLUT11

C

C* CALAULATE MASS MFLOW IN EACH TUBE AND EACH CLASS

MXFLUT12

ZMASSL=ZMASSL+ZMASS(L)*ALPHA(L,J)*(1.-YRICH(L))

MXFLUT13

ZMASSR=ZMASSR+ZMASS(L)*ALPHA(L,J)*YRICH(L)

MXFLUT14
MXFLUT15

301 CONTINUE

MXFLUT16

C

C* CHECK FOR ERROR IN MASS FLOW CALCULATION

MXFLUT17

ERROR(J)=ZMASSR+ZMASSL-ZMASSH(J)

MXFLUT18

XFRHAT(NF,J)=(FUELL+FUFLR)/(ZMASSL+ZMASSR)

MXFLUT19
MXFLUT20

C

C* CALAULATE INTERIM MIXTURE RATION

MXFLUT21

HETRO=ZMASSL/GT,0.0

MXFLUT22
MXFLUT23

FLPRIM = 0.0

MXFLUT24

IF (HETRO) FLPRIM = FUELL / ZMASSL

MXFLUT25

FRPRIM =FUFLR /ZMASSR

MXFLUT26

C

MXFLUT27

C* CALCULATE INTERIM MASS FRACTION OF RICH SPECIES

MXFLUT28

YINTRM=ZMASSR/(ZMASSR+ZMASSL)

MXFLUT29

C

MXFLUT30

C* CALCULATE FINAL MIXTURE RATIOS AT NEW X-STATION

MXFLUT31

FOAIR(JLFAN,J,LNEW)=FLPRIM

MXFLUT32

SMALLF=FOAIR(JLEAN,J,LNEW)-XFRHAT(NF,J)

MXFLUT33

GGG = AMINI(GHAT(J), YINTRM*(FRPRIM-XFRHAT(NF,J))*2+

MXFLUT34

* (1.-YINTRM)*SMALLF**2)

MXFLUT35

FOAIR(JRICH,J,LNEW)= XFRHAT(NF,J)-GGG/SMALLF

MXFLUT36

C

MXFLUT37

C* CALCULATE FRACTION OF RICH SPECIES AT NEW X-STATION

MXFLUT38

YPRIME(J)=-SMALLF/(FOAIR(JRICH,J,LNEW)-FOAIR(JLEAN,J,LNEW))

MXFLUT39

C

MXFLUT40

C* ALPHA(J,L) IS THE FRACTION OF MASS IN TUBE 'J' AT STATION=X, WHICH

MXFLUT41

C* IN TUBE 'L' AT STATION X+DX

MXFLUT42

C

MXFLUT43

C* UPDATE CONCENTRATIONS AT NEW X-STATION

MXFLUT44

C

MXFLUT45

K=J

MXFLUT46

C

MXFLUT47

C* **YYYY** IS PERCENT OF MASS FLOW IN LEAN TUBE THAT IS POURED INTO RICH

MXFLUT48

YYYY=0.0

MXFLUT49

IF(YINTRM .NE. 1.0) YYYY=(YPRIME(K)-YINTRM)/(1.0-YINTRM)

MXFLUT50

```

IF(YYYY,LT,0.0) YYYY=0.0
C
DO 400 I=1,NCNOC
  CONCRH=0
  CONCLN=0
  DO 397 KDUM=1,NSPECI
    C
    C* THIS EQUATION ASSUMES CNOC=LBM I/LBM MIX--NOT LBM I/MOLE MIX
    CONCLN=CONCLN+CNOC(I,JLEAN,KDUM,LOLD)*ALPHA(KDUM,K)*(1.-
    YRICH(KDUM))*ZMASS(KDUM)
    CONCRH=CONCRH+CNOC(I,JRICH,KDUM,LOLD)*ALPHA(KDUM,K)*YRICH(
    KDUM)*ZMASS(KDUM)
    IF(T,EQ,16) TFR(KDUM)=CNOC(I,JRICH,KDUM,LOLD)*ALPHA(KDUM,K)*
    YRICH(KDUM)*ZMASS(KDUM)
    397 CONTINUE
    CONCRH=CONCRH+CONCLN*YYYY
    IF(T,EQ,16) TFL=CONCLN*YYYY
    CNOC(I,JRICH,K,LNEW)=CONCRH/(YPRIME(K)*(ZMASSL+ZMASSR))
    CNOC(I,JLEAN,K,LNEW)=CNOC(I,JLEAN,K,LOLD)
    IF(METRO)CNOC(I,JLEAN,K,LNEW)=CONCLN/ZMASSL
    400 CONTINUE
  C
  ENTHLN=0
  ENTHRH=0
  DO 410 KDUM=1,NSPECI
    FNTHLN=ENTHLN+(ENTH(JLEAN,KDUM,LOLD)+U(KDUM,LOLD)**2/50075.614)
    *ALPHA(KDUM,K)*(1.0-YRICH(KDUM))*ZMASS(KDUM)
    410 ENTHRH=ENTHRH+(ENTH(JRICH,KDUM,LOLD)+U(KDUM,LOLD)**2/50075.614)
    *ALPHA(KDUM,K)*YRICH(KDUM)*ZMASS(KDUM)
    FNTHRH=ENTHRH+FNTHLN*YYYY
    ENTH(JRICH,K,LNEW)=ENTHRH/(YPRIME(K)*(ZMASSL+ZMASSR))
    *U(K,LNEW)**2/50075.614
    FNTH(JLEAN,K,LNEW)=FNTH(JLEAN,K,LOLD)
    IF(METRO)FNTH(JLEAN,K,LNEW)=FNTHLN/ZMASSL
    *U(K,LNEW)**2/50075.614
    TIM2 = TAU(K,2)
    CALL HYPARH(K)
    TIM1(K)=TAU(K,2)
  C
  C CALCULATE H/C INCIPIENCIES
  411 CONTINUE
  XC3=3.*DX
  IF(XCC,LE,XC3.AND,PDUH(12),NE,0.)CALL TABPRT(5HZMASS,ZMASS,12,10)
  IF(XCC,LE,XC3.AND,PDUH(12),NE,0.)CALL TABPRT(5HYRICH,YRICH,12,10)
  IF(XCC,LE,XC3.AND,PDUH(12),NE,0.)CALL TABPRT(6HYPRIME,YPRIME,12,10)
  * )
  IF(XCC,LE,XC3.AND,PDUH(12),NE,0.)CALL TABPRT(4HGAT,GAT,12,10)
  IF(XCC,LE,XC3.AND,PDUH(12),NE,0.)CALL TABPRT(5HFOATR,FOATR,24,10)
  IF(XCC,LE,XC3.AND,PDUH(12),NE,0.)CALL TABPRT(5HALPHA,ALPHA,144,10)
  IF(PDUH(14),NE,0..AND,XCC,LE,PDUH(14))

```

MXFLUT51
 MXFLUT52
 MXFLUT53
 MXFLUT54
 MXFLUT55
 MXFLUT56
 MXFLUT57
 MXFLUT58
 MXFLUT59
 MXFLUT60
 MXFLUT61
 MXFLUT62
 MXFLUT63
 MXFLUT64
 MXFLUT65
 MXFLUT66
 MXFLUT67
 MXFLUT68
 MXFLUT69
 MXFLUT70
 MXFLUT71
 MXFLUT72
 MXFLUT73
 MXFLUT74
 MXFLUT75
 MXFLUT76
 MXFLUT77
 MXFLUT78
 MXFLUT79
 MXFLUT80
 MXFLUT81
 MXFLUT82
 MXFLUT83
 MXFLUT84
 MXFLUT85
 MXFLUT86
 MXFLUT87
 MXFLUT88
 MXFLUT89
 MXFLUT90
 MXFLUT91
 MXFLUT92
 MXFLUT93
 MXFLUT94
 MXFLUT95
 MXFLUT96
 MXFLUT97
 MXFLUT98
 MXFLUT99
 MXFLUT00

* CALL TABPRT(6HGCASCHP,ENTH,864,10)	MXFLUT01
IF(PDUM(16).NE.0.) WRITE (6,10) (XFRHAT(NF,J),J=1,NSPECI)	MXFLUT02
C	MXFLUT03
DO 420 K=1,NTUBES	MXFLUT04
420 YRICH(K)=YPRIMF(K)	MXFLUT05
C* FQAIR IS NOT RESET BETWEEN X-STEPS	MXFLUT06
DO 299 L=1,NTUBES	MXFLUT07
FQAIR(JRICH,L,LOLD)=FQAIR(JRICH,L,LNEW)	MXFLUT08
FQAIR(JLEAN,L,LOLD)=FQAIR(JLEAN,L,LNEW)	MXFLUT09
C	MXFLUT10
C	MXFLUT11
U(L,LOLD)=U(L,LNEW)	MXFLUT12
HCINCP(L,LOLD)=HCINCP(L,LNEW)	MXFLUT13
ENTH(JRICH,L,LOLD)=ENTH(JRICH,L,LNEW)	MXFLUT14
ENTH(JLEAN,L,LOLD)=ENTH(JLEAN,L,LNEW)	MXFLUT15
C* MOVE CONCENTRATIONS FROM NEW TO OLD LOCATIONS UNLESS YOU HAVE JUST	CMXFLUT16
C* --SCKP/GCKP--	MXFLUT17
DO 298 I=1,NCONC	MXFLUT18
CONC(I,JRICH,L,LOLD)=CONC(I,JRICH,L,LNEW)	MXFLUT19
298 CONC(I,JLEAN,L,LOLD)=CONC(I,JLEAN,L,LNEW)	MXFLUT20
299 CONTINUE	MXFLUT21
IF(PDUM(13).NE.0.) CALL TABPRT(6HHCINCP,HCINCP,24,10)	MXFLUT22
IF(PDUM(13).NE.0.) WRITE (6,2880) (CONC(16,1,K,2),K=1,12)	MXFLUT23
2880 FORMAT(/2X,5HFUEL2//2X,6F16.8/2X,6F16.8)	MXFLUT24
RETURN	MXFLUT25
1 FORMAT (1X,4HTUB3F,13,2X,12F10.6)	MXFLUT26
2 FORMAT (1X,2F)	MXFLUT27
3 FORMAT (1X,7HTD TUBE,13,1X,12F10.6)	MXFLUT28
4 FORMAT (10X,12(10H SPECIE))	MXFLUT29
5 FORMAT (10X,12(15,5X))	MXFLUT30
6 FORMAT (10X,49HFLOW OF EACH SPECIE IN EACH TUBE AT OLD X-STATJON,	MXFLUT31
1,35X,10H**XFRAC**)	MXFLUT32
7 FORMAT (10X,49HFLOW OF EACH SPECIE IN EACH TUBE AT NEW X-STATJON,	MXFLUT33
1,35X,10H**XFRHATA**)	MXFLUT34
8 FORMAT (10X,36HNOUTION OFR INTERTUBE MASS TRANSFER,1,35X,9H**ALPH	MXFLUT35
1A**)	MXFLUT36
9 FORMAT (12X,12(10H FROM),1,12X,12(10H TUBE),1,12X,12(15,	MXFLUT37
15X))	MXFLUT38
10 FORMAT (1X,35HMEAN FUEL TO GAS RATIO IN EACH TUBE,1,1X,12F10.6,1)	MXFLUT39
END	MXFLUT40

CMXKIND	MAIN SUB,-- MIX. FLUTTER, KINETICS, OUTPUT	MXKIND01
	SUBROUTINE MXKIND	MXKIND02
LOGICAL	LKGCKP,LKSCKP	MXKIND03
COMMON /INDATA/	N,HF,WAR,T2,BETA,T25,FARS,FINOPC,P0,	MXKIND04
*	RCO(11),RCO2(11),RHC(11),RNOX(11),	MXKIND05
*	PT(11),PS(11),BLOC(11),QCO(11)	MXKIND06
REAL	N	MXKIND07
COMMON /JETDAT/	NPTS,RAD(12),TS(12),UU(12),SPV(12),MWT(12),	MXKIND08
*	CP(12),FUEL(12),SPALDG(12),TKE(12),OTHER1(36)	MXKIND09
REAL	MWT	MXKIND10
COMMON /GASCMP/	RICH(12,2),FUEL(2,12,2),ENTH(2,12,2),	MXKIND11
*	CONC(16,2,12,2),HCINCP(12,2),OTHER(192)	MXKIND12
COMMON /GASTMW/	TG(2,12,2),MWIG(2,12,2),TAU(12,2),CPG(2,12,2)	MXKIND13
REAL	MWTG	MXKIND14
COMMON /DPCTRL/	TITLE(20),PRINT(30)	MXKIND15
COMMON /STCTRL/	LSTA,FINAL,CHEMK,FIRSTM,FIRSTC,SC,DXC,NIST,	MXKIND16
*	DUMSI(9)	MXKIND17
LOGICAL	FINAL,FIRSTM,FIRSTC	MXKIND18
INTEGER	CHEMK	MXKIND19
COMMON /CPRINT/	PDUM(20)	MXKIND20
COMMON /TROUHL/	FRR,ERRMAJ,INERR,PRERR	MXKIND21
LOGICAL	FRR,ERRMAJ,INERR,PRERR	MXKIND22
COMMON /CHITS /	RITS, BLANK	MXKIND23
COMMON /CINSAV/	HGCKP(57)	MXKIND24
COMMON /KOUT/	TITLE1(20),DUMKT(37)	MXKIND25
COMMON /CMASS /	DUMASS(48)	MXKIND26
COMMON /CAXIAL/	DUMAX(3)	MXKIND27
COMMON /PORIDE/	PSTA(30)	MXKIND28
COMMON /DUTREM/	KINFIL	MXKIND29
COMMON /COREQN/	COREQ	MXKIND30
LOGICAL	COREQ	MXKIND31
DIMENSION	CLASS(2)	MXKIND32
C		MXKIND33
	NAMLIST/A/ PDUM,CHEMK,PSTA,NIST ,COREQ	MXKIND34
C		MXKIND35
C	CALCULATE FUEL MW AND ADJUST C2H4 THERMO COEFF	MXKIND36
C		MXKIND37
	CALL ADJMC	MXKIND38
C		MXKIND39
C	INITIALIZE	MXKIND40
C		MXKIND41
	KINFIL=40	MXKIND42
	CALL SFTM(1,RITS,PSTA,30,)	MXKIND43
	CALL SETM(3,0,,PDUM,20,HCINCP,24,TAU,24)	MXKIND44
	NIST = 2	MXKIND45
	CHEMK = 1	MXKIND46
	XC = 0.	MXKIND47
	LSTA = 1	MXKIND48
	FIRSTM= .TRUE.	MXKIND49
	FIRSTC= .TRUE.	MXKIND50

CALL ERRDRW(\$300,\$300)	MXKIN051
READ (5,A)	MXKIN052
C	MXKIN053
C CALCULATE MIXTURE FLUTTER	MXKIN054
C	MXKIN055
1 CALL LLINK(4HLMXF)	MXKIN056
LKGCKP= .FALSE.	MXKIN057
LKSCKP= .FALSE.	MXKIN058
CALL MFMAIN	MXKIN059
CALL MOVE(7,TITLE,TITLE1,20,1)	MXKIN060
FIRSTM= .FALSE.	MXKIN061
NK = NPTS	MXKIN062
C	MXKIN063
C MIXTURE FLUTTER DATA STORED AT PRESENT (2) STATION	MXKIN064
C CALCULATE MOLECULAR WEIGHTS	MXKIN065
C	MXKIN066
CALL MOVE(2,MWTG(1,1,2),MWTG(1,1,1),24,1,TG(1,1,2),	MXKIN067
* TG(1,1,1),24,1)	MXKIN068
DO 6 K=1,NK	MXKIN069
DO 5 J=1,2	MXKIN070
C	MXKIN071
C RESET N ATOM COMPOSITION	MXKIN072
C	MXKIN073
IF(CONC(I2,J,K,2).LE. 1.E-8) CONC(I2,J,K,2)=0.	MXKIN074
SUMC = 0.	MXKIN075
DO 4 I=1,16	MXKIN076
4 SUMC = SUMC+CONC(I,J,K,2)	MXKIN077
5 MWTG(J,K,2)= 1./SUMC	MXKIN078
6 CONTINUE	MXKIN079
C	MXKIN080
C CALCULATE STATIC TEMPERATURE FROM ENTHALPY	MXKIN081
C	MXKIN082
10 CALL TFMH(NK)	MXKIN083
C	MXKIN084
IF(PDUM(14).EQ.0.) GO TO 15	MXKIN085
CALL TABPRT(6HGASCP,RICH,912,10)	MXKIN086
CALL TABPRT(6HGASTMW,TG,96,10)	MXKIN087
15 CONTINUE	MXKIN088
C	MXKIN089
C*** SECTION TO CALCULATE TIME HISTORY OF REACTIONS	MXKIN090
C	MXKIN091
DO 60 K=1,NK	MXKIN092
C	MXKIN093
C RECORD STARTING POINT FOR KINETICS STEP (JUST IN CASE)	MXKIN094
C	MXKIN095
REWIND KINFIL	MXKIN096
DATA CLASS /4HRICH,4HLEAF/	MXKIN097
WRITE(KINFIL,1933)K,RICH(K,2),HCINCP(K,2),(TAU(K,L),L=1,2),	MXKIN098
(CLASS(L),FUEL(L,K,2),TG(L,K,2),ENTH(L,K,2),(CONC(IC,L,K,2),	MXKIN099
IC=1,16),L=1,2)	MXKIN000

1933	FORMAT(49H INITIAL CONDITIONS FOR LAST KINETICS CALCULATION,	MXXKIND01
	'60X,11HSTREAMTUBE ,I2,10X/132X/8H YRICH= ,F11.8,10H HCINCP= ,	MXXKIND02
	IF11.8,12H TIME FROM ,E15.8,4H TO ,E15.8,8H SECONDS,36X/(132X/	MXXKIND03
	1X,A4,14H PART FUEL= ,F12.9,5H T= ,F10.4,5H H= ,E15.8,	MXXKIND04
	128H GAS COMPOSITION, MOLES/LB,38X/1X,8E15.8,11X/1X,8E15.8,11X))	MXXKIND05
C		MXXKIND06
	30 GO TO (40,45,50) , CHEMK	MXXKIND07
	50 CALL CHEMB(IBRNCH)	MXXKIND08
C		MXXKIND09
	GO TO (40,45) , IBRNCH	MXXKIND10
C		MXXKIND11
	40 IF(LKGCKP) GO TO 41	MXXKIND12
	CALL LLINK(4HLSCK)	MXXKIND13
	LKGCKP= .TRUE,	MXXKIND14
	LKSCKP= .FALSE.	MXXKIND15
	41 CALL GCKP(K)	MXXKIND16
	IF(ERR) GO TO 205	MXXKIND17
	GO TO 60	MXXKIND18
C		MXXKIND19
	45 IF(LKSCKP) GO TO 46	MXXKIND20
	CALL LLINK(4HLSCK)	MXXKIND21
	LKSCKP= .TRUE,	MXXKIND22
	LKGCKP= .FALSE.	MXXKIND23
	46 CALL SCKP(K)	MXXKIND24
	IF(ERR) GO TO 200	MXXKIND25
	60 CONTINUE	MXXKIND26
	CALL MOVE(1,TAU(1,2),TAU(1,1),12,1)	MXXKIND27
C		MXXKIND28
C	TEST FOR PRINT STATION	MXXKIND29
C		MXXKIND30
	IF(.NOT.FINAL) GO TO 1	MXXKIND31
	CALL LLINK(4HLSFIN)	MXXKIND32
	LKSCKP=.FALSE,	MXXKIND33
	LKGCKP=.FALSE.	MXXKIND34
	CALL SUMUP	MXXKIND35
	FINAL = .FALSE.	MXXKIND36
	LSTA = LSTA+1	MXXKIND37
	IF(PRINT(LSTA) .NE.BITS) GO TO 1	MXXKIND38
	100 RETURN	MXXKIND39
C		MXXKIND40
	200 WRITE (6,201)	MXXKIND41
	201 FORMAT(/2X,25HERROR IN SCKP CALCULATION//)	MXXKIND42
	GO TO 300	MXXKIND43
	205 WRITE (6,206)	MXXKIND44
	206 FORMAT(/2X,25HERROR IN GCKP CALCULATION//)	MXXKIND45
	300 ERR = .TRUE.	MXXKIND46
	IF(.NOT.LKSCKP) CALL LLINK(4HLSCK)	MXXKIND47
	CALL ERROR1	MXXKIND48
	RETURN	MXXKIND49
	END	MXXKIND50

C	ENAMBLK	-NAMRLK-	BLOCK DATA	(A7N)	NAMBLK01
C			BLOCK DATA		NAMBLK02
C			ALPHANUMERIC DATA FOR TESTING AND OUTPUT		NAMBLK03
C			*** FOR GCKP-1	AUGUST 1973 *****	NAMBLK04
C			COMMON / LTUS / LTHM, LDAT		NAMBLK05
C			COMMON / OPTS / TIMIN, TIME, BLNK, AREA, DUM3(4)		NAMBLK06
C			COMMON / SPEC / SNAM(2,3), DUM4(2,25), EFFM(2), BLANK(2)		NAMBLK07
C			COMMON / KOUT / TITLE(20), UNITI, UNITO, DUM6(32), FPS, SI, DUM7		NAMBLK08
C			LOGICAL TAPE UNIT ASSIGNMENTS		NAMBLK09
C			*** (NOT USED IN GCKP-1)***		NAMBLK10
C			DATA LTHM, LDAT/4, 43/		NAMBLK11
C			ALPHANUMERIC DATA		NAMBLK12
C			DATA TIME, AREA/4HTIME, 4HAREA/		NAMBLK13
C			DATA SNAM, EFFM, BLANK/1HV, 1H, 3HRHO, 1H, 1HT, 1H, 1HM, 1H,		NAMBLK14
C			DATA FPS, SI/3HFPS, 2HSI/		NAMBLK15
C			DATA TITLE/		NAMBLK16
C			* 4H , 4H , 4HKASK, 4HANS , 4HEYPE, 4HRTME,		NAMBLK17
C			* 4HNT , 4H POS, 4HT-FL, 4HAME , 4HREAC, 4HTION,		NAMBLK18
C			* 4HS IN, 4H LEA, 4HN C2, 4HHQ-A, 4HIR F, 4HLAME,		NAMBLK19
C			* 4H , 4H /		NAMBLK20
C			DATA UNITI,UNITO/3HFPS,3HFPS/		NAMBLK21
C			DATA BLNK/1H /		NAMBLK22
C			DATA TIMIN/4HTIME/		NAMBLK23
C			END		NAMBLK24
					NAMBLK25
					NAMBLK26
					NAMBLK27
					NAMBLK28
					NAMBLK29
					NAMBLK30
					NAMBLK31
					NAMBLK32
					NAMBLK33
					NAMBLK34
					NAMBLK35
					NAMBLK36
					NAMBLK37
					NAMBLK38

CNOX2B	INTEGRATE NO RATE EQUATIONS	NOX2B001
	SUBROUTINE NOX2B(P,DTIME,XNOI)	NOX2B002
	COMMON /PSF3 / FJA,BETS,TP,X(16),DHQDMW,TEQ,BEQ,XMWT,HNEQ	NOX2B003
	EQUIVALENCE (X1,X(1)),(X2,X(2)),(X3,X(3)),(X4,X(4)),(X5,X(5)),	NOX2B004
*	(X7,X(7)),(X8,X(8)),(X9,X(9)),(X10,X(10)),	NOX2B005
*	(X11,X(11)),(X12,X(12))	NOX2B006
	COMMON /GHSC / F(25),HH(25),SR(25),CPZ(25),DCPR(25)	NOX2B007
	COMMON /NOXTRA/ DNOXDT	NOX2B008
	REAL MWT,K1,K2,K3,KR1,KR2,KR3	NOX2B009
	DIMENSION RK(4)	NOX2B010
	DATA C2/.01603286/,K3/40.E+12/	NOX2B011
C		NOX2B012
C	TK = TP / 1.8	NOX2B013
	RHOM = 144.*P*C2/(1545.32*TP)	NOX2B014
	N = 1	NOX2B015
C		NOX2B016
C	DETERMINE EQUILIBRIUM CONSTANTS	NOX2B017
C		NOX2B018
	1 CALL THRM(TK,1.)	NOX2B019
	EQ1 = EXP(F(9)+F(2)-F(11)-F(12))	NOX2B020
	EQ2 = EXP(F(12)+F(4)-F(11)-F(2))	NOX2B021
	EQ3 = EXP(F(12)+F(5)-F(11)-F(1))	NOX2B022
	K1 = EQ1*3.10E13*EXP(-168.1874/TK)	NOX2B023
	K2 = 6.40E09*TK*EXP(-3147.22/TK)	NOX2B024
	KR1 = K1/EQ1	NOX2B025
	KR2 = K2/EQ2	NOX2B026
	KR3 = K3/EQ3	NOX2B027
	TIMEK = DTIME	NOX2B028
C		NOX2B029
C	COMPUTE DXNO/DT FOR RK INTEGRATION	NOX2B030
C		NOX2B031
	XNOT = XNOI	NOX2B032
	5 R1 = 2.*K1*RHOM*(X9*X2*(K2*X4+K3*X5)-XNOT**2/EQ1*	NOX2B033
	* (KR2*X2+KR3*X1))	NOX2B034
	R2 = KR1*XNOT+K2*X4+K3*X5	NOX2B035
C		NOX2B036
	15 RK(N) = R1/R2	NOX2B037
	GO TO (16,16,20,40) , N	NOX2B038
	16 XNOT = XNOI+.5*RK(N)*TIMEK	NOX2B039
	GO TO 25	NOX2B040
	20 XNOT = XNOI+RK(N)*TIMEK	NOX2B041
	25 N = N+1	NOX2B042
	GO TO 5	NOX2B043
	40 DNOXDT= (RK(1)+2.*(RK(2)+RK(3))+RK(4))/6.	NOX2B044
	XNOI = XNOI+DNOXDT*TIMEK	NOX2B045
	X12 = (K1*X9*X2+KR2*XNOT*X2+KR3*XNOT*X1)/R2	NOX2B046
C		NOX2B047
C		NOX2B048
	100 RETURN	NOX2B049
		NOX2B050

END

NOX28051

CNWN	CONVERT JETMIX PROFILES TO FIXED-NUMBER STREAM NET	NNNET001
C	SUITABLE FOR MIXING-REACTION CALCULATIONS	NNNET002
C		NNNET003
	SUBROUTINE NEWNET	NNNET004
C		NNNET005
	INTEGER AGAIN	NNNET006
	REAL MWT,JX,MWTO	NNNET007
	DIMENSION GJX(50),WKCJX(50,12),UDF(12)	NNNET008
C		NNNET009
	COMMON /CAGAIN/ AGAIN	NNNET010
	COMMON /CINPJT/ DIAJ,DUM3(64)	NNNET011
	COMMON /JETDAT/ N,RAD(12),TO(12),U(12),SPVO(12),MWTO(12),CP0(12),	NNNET012
	FO(12),G(12),TKE0(12),OTHR0(36)	NNNET013
	COMMON /CCHCK/ DUM4(12),WKTOT(12),DUM5(12)	NNNET014
	COMMON /CNWNET/ WFCJX(50),DUM6(50),WFCJUM(12),DUM7(50),PSIJX(50)	NNNET015
	COMMON /CSPEC/ M,NF,DX	NNNET016
	COMMON /CINPJT/ XX(5),RJR(5,50),UJR(5,50),RHOJR(5,50),	NNNET017
	XKJR(5,50,13),PSIJR(5,50),GJR(5,50),NJJ(5),NTI	NNNET018
	COMMON /CAXIAL/ DUMB,DUM9,X	NNNET019
	COMMON /CLOCAL/ RJX(50),UJX(50),DUM10(50),XKJX(50,13),NJ	NNNET020
	COMMON /CMXELL/ W(12,12),WHAT(13,12),ALPHA(12,12)	NNNET021
	COMMON /CMASS/ WTOLD(12),WTOT(12),DUM11(24)	NNNET022
	COMMON /GASTHW/ DUM1(96),TAU(12,2),DUM2(48)	NNNET023
	COMMON /CPRINT/ PDUM(20)	NNNET024
	DIMENSION CFACSP(13)	NNNET025
	EQUIVALENC (CORFAC,CFACSP(13))	NNNET026
C		NNNET027
C	FIRST TIME THRU, DERIVE CONTINUITY CHECK PARAMETERS	NNNET028
C		NNNET029
	IF(AGAIN.EQ.(-1))GO TO 12	NNNET030
C		NNNET031
C	REPEAT PREJET CALCULATION OF INITIAL TUBE AREAS	NNNET032
C		NNNET033
C	SET UP SPECIE FLOW MATRIX FOR INITIAL STEP PROFILE	NNNET034
C		NNNET035
	CALL SETM(2,0.,WHAT,156,WTOT,12)	NNNET036
	RTI2=0.0	NNNET037
	WFCJUM(1)=0.0	NNNET038
	DO 10 J=1,N	NNNET039
	RTD2=0.5*(RAD(I)**2+RAD(J+1)**2)	NNNET040
	IF(J.EQ.N)RTD2=RAD(N)**2	NNNET041
	ATO=3.14156*(RTD2-RTI2)	NNNET042
	RTI2=RTD2	NNNET043
C		NNNET044
C	TOTAL FLOW OF EACH SPECIE AND	NNNET045
C	CUMULATIVE FUEL FLOW IN EACH TUBE.	NNNET046
C		NNNET047
	WKTOT(J)=ATO*U(J)/SPVO(I)	NNNET048
	WTOT(J)=WKTOT(J)	NNNET049
	WHAT(J,J)=WKTOT(J)	NNNET050

WF=F0(J)*WKTOT(J)	NNNET051
WFCUM(J+1)=WFCUM(J)+WF	NNNET052
10 CONTINUE	NNNET053
C SET SPECIES FLOW MATRIX AT OLD STATION EQUAL TO PREVIOUSLY	NNNET054
C DERIVED FLOW MATRIX	NNNET055
C	NNNET056
12 DO 14 J=1,M	NNNET057
14 CALL MOVE(1,WHAT(1,J),W(1,J),12,1)	NNNET058
C	NNNET059
C STORE TOTAL FLOW IN EACH TUBE AT OLD STATION	NNNET060
C	NNNET061
C CALL MOVE(1,WTOT,WTOLD,12,1)	NNNET062
C	NNNET063
C INTERPOLATE WITH AXIAL DISTANCE IN JETMIX PROPERTY TABLE	NNNET064
C FROM READT TO ESTABLISH JETMIX PROFILE AT X.	NNNET065
C FIRST, RADIUS (RJX) AND VELOCITY (UJX)	NNNET066
C	NNNET067
20 NJ=NJJ(1)	NNNET068
DO 50 JJ=1,NJ	NNNET069
CALL LSPFIT(XX,RJR(1,JJ),NIT,X,RJX(JJ),1,0)	NNNET070
CALL LSPFIT(XX,UJR(1,JJ),NIT,X,UJX(JJ),1,0)	NNNET071
C	NNNET072
C GET MOLE FRACTION OF SPECIE K AT POINT JJ (XKJX) AND	NNNET073
C MEAN MOLECULAR WEIGHT	NNNET074
C	NNNET075
MWTJX=0.0	NNNET076
DO 30 K=1,M	NNNET077
CALL LSPFIT(XX,XKJR(1,JJ,K),NIT,X,XKJX(JJ,K),1,0)	NNNET078
30 MWTJX=MWTJX+XKJX(JJ,K)*MWT0(K)	NNNET079
C	NNNET080
C CONVERT MOLE FRACTIONS TO MASS FRACTIONS AND	NNNET081
C COMPUTE FUEL MASS FRACTION (STORE IN XKJX(JJ,M+1))	NNNET082
C	NNNET083
XKJX(JJ,NF)=0.0	NNNET084
DO 40 K=1,M	NNNET085
XKJX(JJ,K)=XKJX(JJ,K)*MWT0(K)/MWTJX	NNNET086
40 XKJX(JJ,NF)=XKJX(JJ,NF)+XKJX(JJ,K)*F0(K)	NNNET087
C	NNNET088
C GET SPALDING HETEROGENEITY FACTOR GJX	NNNET089
C	NNNET090
CALL LFIT(XX,GJR(1,JJ),NIT,X,GJX(JJ),1,0)	NNNET091
50 CONTINUE	NNNET092
C	NNNET093
C GET VALUE OF STREAM FUNCTION (PSIJX) AT POINT JJ, THEN	NNNET094
C CUMULATIVE SPECIE FLOWS (WKCJX) AND FUEL FLOW (WFCJX)	NNNET095
C	NNNET096
PSIJX(1)=0.0	NNNET097
WFCJX(1)=0.0	NNNET098
DO 55 K=1,M	NNNET099
	NNNET100

55	WKCJX(1,K)=0.0	NWNET101
	DO 65 JJ=2,NJ	NWNET102
	CALL LFIT(YX,PSIJR(1,JJ),NTI,X,PSIJX(JJ),1,0)	NWNET103
	DWJX=PSIJX(JJ)-PSIJX(JJ-1)	NWNET104
	DO 60 K=1,M	NWNET105
60	WKCJX(JJ,K)=WKCJX(JJ-1,K)+0.5*(XKJX(JJ-1,K)+XKJX(JJ,K))*DWJX	NWNET106
65	WFCJX(JJ)=WFCJX(JJ-1)+0.5*(XKJX(JJ-1,NF)+XKJX(JJ,NF))*DWJX	NWNET107
C		NWNET108
C	SCALE FUEL FLOWS TO MATCH JET EXIT	NWNET109
	CORFAC=WFCUM(M)/WFCJX(NJ)	NWNET110
	DO 70 JJ=1,NJ	NWNET111
70	WFCJX(JJ)=WFCJX(JJ)*CORFAC	NWNET112
C		NWNET113
C	TABULATE RADIUS AND SPECIE FLOWS AGAINST FUEL FLOW.	NWNET114
C	INTERPOLATE TO GET REACTION TUBE BOUNDARIES AND SPECIE FLOWS.	NWNET115
C		NWNET116
	CALL LSPFIT(WFCJX,RJX,NJ,WFCUM,RAD,M,0)	NWNET117
	DO 75 K=1,M	NWNET118
75	RAD(K)=SQRT(RAD(K))	NWNET119
	DO 90 K=1,M	NWNET120
	CALL LSPFIT(WFCJX,WKCJX(1,K),NJ,WFCUM,WHAT(1,K),M,0)	NWNET121
C		NWNET122
C	CONVERT CUMULATIVE SPECIE FLOWS TO INCREMENTAL FLOWS IN EACH TUBE	NWNET123
C	INVERT SPECIE FLOW MATRIX USING ALPHA FOR TEMP. STORAGE.	NWNET124
C		NWNET125
	DO 80 J=1,N	NWNET126
	WHAT(J,K)=WHAT(J+1,K)-WHAT(J,K)	NWNET127
80	ALPHA(K,J)=WHAT(J,K)	NWNET128
90	CONTINUE	NWNET129
	DO 100 J=1,N	NWNET130
100	CALL MOVE(1,ALPHA(1,J),WHAT(1,J),12,1)	NWNET131
C		NWNET132
C	INTEGRATE VELOCITY AND HETEROGENEITY ACROSS EACH REACTION TUBE	NWNET133
C	TO GET MEAN VALUE.	NWNET134
C		NWNET135
	UDF(1)=UJX(1)	NWNET136
	G(1)=GJX(1)	NWNET137
	CALL LFIT(WFCJX,UJX,NJ,WFCUM,UDF,M,-1)	NWNET138
	CALL LFIT(WFCJX,GJX,NJ,WFCUM,G,M,-1)	NWNET139
	DO 110 J=1,N	NWNET140
	DWF=WFCUM(J+1)-WFCUM(J)	NWNET141
	U(J)=(UDF(J+1)-UDF(J))/DWF	NWNET142
110	G(J)=AMAX1(0.0,(G(J+1)-G(J))/DWF)	NWNET143
	R(M)=2	NWNET144
C		NWNET145
C	CORRECT SPECIE FLOWS IN EACH TUBE SO THAT TOTAL FLOW OF EACH SPECIE	NWNET146
C	AGREES WITH INITIAL VALUE. AIR FLOW LEFT AS IS.	NWNET147
C		NWNET148
	DO 140 K=1,N	NWNET149
		NWNET150

WHAT(K,M)=0.0	NWNET151
SUMWK=0.0	NWNET152
DO 120 J=1,M	NWNET153
120 SUMWK=SUMWK+WHAT(K,J)	NWNET154
CFACSP(K)= WKTOT(K)/SUMWK	NWNET155
DO 130 J=1,M	NWNET156
130 WHAT(K,J)= WHAT(K,J)*CFACSP(K)	NWNET157
140 CONTINUE	NWNET158
C	NWNET159
C GET TOTAL FLOW IN EACH TUBE (WTOT).	NWNET160
C	NWNET161
WHAT(M,M)=0.0	NWNET162
WTOT(M)= 0.	NWNET163
DO 151 J=1,N	NWNET164
WTOT(J)=0.0	NWNET165
DO 150 K=1,M	NWNET166
150 WTOT(J)=WTOT(J)+WHAT(K,J)	NWNET167
151 WTOT(M)= WTOT(M)+WTOT(J)	NWNET168
C DEFINE OUTER TUBE FOR ENTRAINED AIR (N IS SPECIE FLOW	NWNET169
C MATRIX AT OLD STATION).	NWNET170
C	NWNET171
W(M,M)=0.0	NWNET172
DO 160 J=1,N	NWNET173
160 W(M,M)=W(M,M)+WHAT(M,J)-W(M,J)	NWNET174
IF(PDUM(16).EQ.0.) GO TO 152	NWNET175
CALL TABPRT(6HCFACSP,CFACSP,13,10)	NWNET176
CALL TABPRT(5HPSIJR,PSIJR,250,10)	NWNET177
CALL TABPRT(5HPSIJX,PSIJX,NJ,10)	NWNET178
CALL TABPRT(4HRJX2,RJX,NJ,10)	NWNET179
CALL TABPRT(3HRAD,RAD,M,10)	NWNET180
CALL TABPRT(5HWFUM,WFCUM,M,10)	NWNET181
CALL TABPRT(4HWTOT,WTOT,M,10)	NWNET182
C	NWNET183
152 IF(AGAIN.NE. (-1)) GO TO 180	NWNET184
C	NWNET185
C REACTION TIME (TAU) UP TO NEW STATION (SECONDS).	NWNET186
C	NWNET187
DXFT=DX*DIAJ/12.0	NWNET188
DO 170 J=1,N	NWNET189
170 TAU(J,2)=TAU(J,2)+DXFT/U(J)	NWNET190
C	NWNET191
180 AGAIN=1	NWNET192
RETURN	NWNET193
END	NWNET194

CNWPSI	REDISTRIBUTION OF STREAM FUNCTION FROM JETMIX	NWPSI001
C	BASED ON LAST KNOWN FLOWS IN NEWNET TUBES.	NWPSI002
C		NWPSI003
C	SUBROUTINE NEWPSI(PSINew,NN)	NWPSI004
C		NWPSI005
C	DIMENSION PSINew(50)	NWPSI006
C	COMMON/CMASS/DUM1(12),FLOW(12),DUM2(24)	NWPSI007
C	COMMON/JETDAT/NT,RAD(12),T(12),U(12),SPV(12),DUM3(96)	NWPSI008
C		NWPSI009
C	BASIC NUMBER OF PSI'S PER NEWNET TUBE AND LAST TUBE	NWPSI010
C	NOT TO RECEIVE EXTRA PSI	NWPSI011
C		NWPSI012
C	NPTB=(NN-1)/NT	NWPSI013
C	NTX1=NT-MOD((NN-1),NT)	NWPSI014
C		NWPSI015
C	FIRST PSI IS ALWAYS CENTERLINE	NWPSI016
C		NWPSI017
C	N=1	NWPSI018
C	PSINew(N)=0.0	NWPSI019
C		NWPSI020
C	STEP FROM TUBE TO TUBE	NWPSI021
C		NWPSI022
C	DO 20 I=1,NT	NWPSI023
C		NWPSI024
C	FRACTION OF FLOW IN THIS TUBE	NWPSI025
C		NWPSI026
C	DPSI=FLOW(I)/FLOW(NT+1)	NWPSI027
C		NWPSI028
C	NUMBER OF PSI'S HERE AND INCREMENT	NWPSI029
C		NWPSI030
C	NPT=NPTB	NWPSI031
C	IF(I.GT,NTX1)NPT=NPTB+1	NWPSI032
C	DPSI=DPSI/ELQAT(NPT)	NWPSI033
C		NWPSI034
C	EVALUATE PSI'S IN THIS TUBE	NWPSI035
C		NWPSI036
C	DO 10 J=1,NPT	NWPSI037
C	N=N+1	NWPSI038
C	10 PSINew(N)=PSINew(N-1)+DPSI	NWPSI039
C	20 CONTINUE	NWPSI040
C	RETURN	NWPSI041
C		NWPSI042
C	IN THE BEGINNING, THERE ARE NO FLOWS	NWPSI043
C		NWPSI044
C	ENTRY NEWFLO	NWPSI045
C	NT1=NT+1	NWPSI046
C	RT12=0.0	NWPSI047
C	FLOW(NT1)=0.0	NWPSI048
C	DO 30 I=1,NT	NWPSI049
C	RT12=0.5*(RAD(I)**2+RAD(I+1)**2)	NWPSI050

```
IF(I.EQ.NT)RTD2=RAD(NT1)**2  
FLOW(I)=3.14159*(RTD2-RTI2)*U(I)/SPV(I)  
FLOW(NT1)=FLOW(NT1)+FLOW(I)  
30 RTI2=RTD2  
RETURN  
END
```

```
NWPSI051  
NWPSI052  
NWPSI053  
NWPSI054  
NWPSI055  
NWPSI056
```

C	OUTPP	GENERAL OUTPUT, UNIT CONVERSIONS, ETC	OUTPP001
		SUBROUTINE OUTP	OUTPP002
C			OUTPP003
C		OUTPUT CAN BE GIVEN IN (1) INTERNAL (CGS) UNITS, (2) FPS UNITS	OUTPP004
C		(3) SI UNITS	OUTPP005
C			OUTPP006
C		LOGICAL ALLM1, CONC, DEBUG, FXCHR, NEXT, RHOCON, TCON, BRIEF	OUTPP007
C			OUTPP008
C		REAL MDOT, TVAR, N, M, MW, MIXMW, M2, MACH, ISUBM	OUTPP009
C			OUTPP010
C		DIMENSION SPNM(2,27), PRX(25), PRX(30), XXH(30), ESP(2,25), DUM1(41)	OUTPP011
C			OUTPP012
		COMMON/OPTS/VERSI, TIMEV, VERSA, AREAV, FLIM, TCON, RHOCON, IPRCON	OUTPP013
		COMMON/COND/DVAR, AREA, MDOT, P, IVAR, V, RHO, T, SIGMA(25), LS, LSP3, NEXT	OUTPP014
		COMMON/SINT/HMIN, HINI, HN, HNP1, HMAX, NH, AVH, EMAX, EPRN, JCV, KOUNT, EPRP	OUTPP015
		COMMON/XOUT/TITLE(20), UNIT1, UNIT2, CONC, FXCHR, DELH(30), FPS, SI, DEBUG	OUTPP016
		COMMON/REFC/LSR(4,30), XX(30), RATE(30), LKER(30), DLKER(30), NH(30), LR	OUTPP017
		COMMON/RRAT/A(30), N(30), FACT(30), B(30), M(25,30), ALLM1	OUTPP018
		COMMON/AFUN/CN(4), ITPS2, LSUBM, ETA, D, VISC, BETAL	OUTPP019
		COMMON/SPEC/SHAM(2,30), MW(25), W(25), STOTC(25,30), OMEGA(25,30)	OUTPP020
		COMMON/XVSA/XTR(40), ATH(40), NTH, XU, AU(2), CX(4), BRIFF, TMLMT	OUTPP021
		COMMON/NECF/PR, MIXMW, M2, GAMMA, TCPR, R	OUTPP022
		COMMON/PORF/PK(28), GK(28), RK(28), E(28)	OUTPP023
		COMMON/DEFN/F(28), ALPHA(28), BETA(28,28)	OUTPP024
		COMMON/SKTP/NEGL(25), I1, I2, IT	OUTPP025
		COMMON/SHSC/GRT(25), HRT(25), SR(25), CDR(25), DCPR(25)	OUTPP026
		COMMON/SABS/S1, AA, HB, S2, DA, D2A, DTERM, RHO	OUTPP027
C			OUTPP028
		EQUIVALENC (SPNM, SHAM(1,4)), (BLANK, SPNM(1,27))	OUTPP029
		EQUIVALENC (PRX(1), XXH(1)), (DUM1(1), ESP(1,1))	OUTPP030
C			OUTPP031
		ENTRY OUT1	OUTPP032
C			OUTPP033
C	** TITLE PAGE		OUTPP034
		IF (VERSI .EQ. TIMEV) GO TO 98	OUTPP035
		I = 2	OUTPP036
		GO TO 99	OUTPP037
		98 I = 4	OUTPP038
		99 IF (VERSA .EQ. AREAV) I = I - 1	OUTPP039
		GO TO (100,200,300,400), I	OUTPP040
		100 WRITE (6,101)	OUTPP041
		101 FORMAT (1H1,14X,21HDISTANCE-AREA VERSION)	OUTPP042
		GO TO 3	OUTPP043
		200 WRITE (6,201)	OUTPP044
		201 FORMAT (1H1,12X,25HDISTANCE-PRESSURE VERSION)	OUTPP045
		GO TO 3	OUTPP046
		300 WRITE (6,301)	OUTPP047
		301 FORMAT (1H1,16X,17HTIME-AREA VERSION)	OUTPP048
		GO TO 3	OUTPP049
		400 WRITE (6,401)	OUTPP050

401	FORMAT (1H1,14X,21HTIME-PRESSURE VERSION)	OUTPP051
3	WRITE (6,102) (TITLE(I),I=1,20)	OUTPP052
102	FORMAT (1H+,49X,33HGENERAL CHEMICAL KINETICS PROGRAM,11X,26HNASA	OUTPP053
	*FWIS RESEARCH CENTER///26X,20A4///9X,8HREACTION,31X,8HREACTION,	OUTPP054
	* 38X,23HREACTION RATE VARIABLES/10X,6HNUMBER,74X,1HA,16X,1HN,9X,	OUTPP055
	* 10HACTIVATION/119X,6HENERGY)	OUTPP056
	KEY=1	OUTPP057
	IF(BRIFF)WRITE(3)KEY,TITLE,UNITO,FPS,SI,LS,SPNM	OUTPP058
C		OUTPP059
C	PRINT REACTION INFORMATION	OUTPP060
	DO 6 J=1,LR	OUTPP061
	N1 = LSR(1,J)	OUTPP062
	N2 = LSR(2,J)	OUTPP063
	N3 = LSR(3,J)	OUTPP064
	N4 = LSR(4,J)	OUTPP065
	WRITE (6,103) J,(SPNM(I,N2),I=1,2),(SPNM(I,N3),I=1,2),A(J),N(J),	OUTPP066
	* EACT(J)	OUTPP067
103	FORMAT (12X,12,27X,2A4,2X,1H=,2X,2A4,23X,1PE12.5,5X,0PF10,4,5X,	OUTPP068
	* F10,2)	OUTPP069
	IF (N1 .GT. 0) GO TO 5	OUTPP070
	IF (N1 .LT. 0) GO TO 4	OUTPP071
	N1 = 26	OUTPP072
	GO TO 205	OUTPP073
4	WRITE (6,105) (SPNM(I,N4),I=1,2)	OUTPP074
105	FORMAT (1H+,63X,1H+,2X,2A4)	OUTPP075
	GO TO 6	OUTPP076
5	IF (N4 .GT. 0) GO TO 205	OUTPP077
	IF (N4 .LT. 0) GO TO 204	OUTPP078
	N4 = 26	OUTPP079
	GO TO 205	OUTPP080
204	WRITE (6,1105) (SPNM(I,N1),I=1,2)	OUTPP081
1105	FORMAT (1H+,27X,2A4,2X,1H+)	OUTPP082
	GO TO 6	OUTPP083
205	WRITE (6,104) (SPNM(I,N1),I=1,2),(SPNM(I,N4),I=1,2)	OUTPP084
104	FORMAT (1H+,27X,2A4,2X,1H+,25X,1H+,2X,2A4)	OUTPP085
C		OUTPP086
C	CONVERT ACTIVATION ENERGY TO B-FACTOR	OUTPP087
	6 B(J) = EACT(J)/1.987165	OUTPP088
C		OUTPP089
	IF (.NOT. ALLM1) GO TO 7	OUTPP090
	WRITE (6,106)	OUTPP091
106	FORMAT (///51X,29HALL THIRD BODY RATIOS ARE 1.0)	OUTPP092
	GO TO 13	OUTPP093
C		OUTPP094
7	WRITE (6,107)	OUTPP095
107	FORMAT (///41X,50HALL THIRD BODY RATIOS ARE 1.0 EXCEPT THE FOLLOWING	OUTPP096
	*NG//)	OUTPP097
	K = 0	OUTPP098
	DO 12 I=1,13	OUTPP099
	DO 12 J=1,1R	OUTPP100

IF (M(I,J),EQ,1) GO TO 12	OUTPP101
K = K + 1	OUTPP102
IF (K,EO,5) K = 1	OUTPP103
GO TO (8,9,10,11),K	OUTPP104
8 WRITE (6,108) (SPNM(K,I),K=1,2),J,M(I,J)	OUTPP105
108 FORMAT (5X,2HM(.2A4,1H,,12,3H) =,F10.5)	OUTPP106
GO TO 12	OUTPP107
9 WRITE (6,109) (SPNM(K,I),K=1,2),J,M(I,J)	OUTPP108
109 FORMAT (1H+,36X,2HM(.2A4,1H,,12,3H) =,F10.5)	OUTPP109
GO TO 12	OUTPP110
10 WRITE (6,110) (SPNM(K,I),K=1,2),J,M(I,J)	OUTPP111
110 FORMAT (1H+,68X,2HM(.2A4,1H,,12,3H) =,F10.5)	OUTPP112
GO TO 12	OUTPP113
11 WRITE (6,111) (SPNM(K,I),K=1,2),J,M(I,J)	OUTPP114
111 FORMAT (1H+,100X,2HM(.2A4,1H,,12,3H) =,F10.5)	OUTPP115
12 CONTINUE	OUTPP116
C	OUTPP117
13 IF (VERST,EO,TIMEV) GO TO 14	OUTPP118
WRITE (6,112) HMIN,HMAX,HINT,EMAX	OUTPP119
112 FORMAT (///56X,20HINTEGRATION CONTROL S//15X,17HMINIMUM STEP SIZE,	OUTPP120
* E14.5,3H CM,33X,17HMAXIMUM STEP SIZE,F14.5,3H CM//15X,17HINITIAL	OUTPP121
* STEP SIZE,F14.5,3H CM,33X,22HMAXIMUM RELATIVE ERROR,F10.5)	OUTPP122
GO TO 15	OUTPP123
14 WRITE (6,113) HMIN,HMAX,HINT,EMAX	OUTPP124
113 FORMAT (///56X,20HINTEGRATION CONTROL S//15X,17HMINIMUM STEP SIZE,	OUTPP125
* E14.5,4H SEC,32X,17HMAXIMUM STEP SIZE,E14.5,4H SEC//15X,17HINITIAL	OUTPP126
* L STEP SIZE,F14.5,4H SEC,32X,22HMAXIMUM RELATIVE ERROR,F10.5)	OUTPP127
C	OUTPP128
C ** SECOND PAGE	OUTPP129
15 WRITE (6,114)	OUTPP130
114 FORMAT (1H,50X,31H** ASSIGNED VARIABLE PROFILE **//)	OUTPP131
GO TO (16,18,19,19,20),ITPSZ	OUTPP132
C	OUTPP133
16 GO TO (116,216,316,416),IPRCOD	OUTPP134
C ASSIGNED VARIABLE TABLE	OUTPP135
116 WRITE (6,117) XU,AU	OUTPP136
117 FORMAT (34X,64HTHE AREA IS CALCULATED BY INTERPOLATION FROM THE FOD	OUTPP137
* LLONGING TABLE//36X,7HSTATION,10X,17HAXIAL DISTANCE (,A2,1H),10X,	OUTPP138
* 7HAREA (,A4,A1,1H))	OUTPP139
GO TO 516	OUTPP140
216 WRITE (6,217) XU,AU	OUTPP141
217 FORMAT (32X,68HTHE PRESSURE IS CALCULATED BY INTERPOLATION FROM THE	OUTPP142
* F FOLLOWING TABLE//36X,7HSTATION,10X,17HAXIAL DISTANCE (,A2,1H),	OUTPP143
* 9X,11HPRESSURE (,2A4,1H))	OUTPP144
GO TO 516	OUTPP145
316 WRITE (6,317) AU	OUTPP146
317 FORMAT (34X,64HTHE AREA IS CALCULATED BY INTERPOLATION FROM THE FOD	OUTPP147
* LLONGING TABLE//36X,7HSTATION,14X,11HTIME (SEC),16X,7HAREA (,A4,	OUTPP148
* A1,1H))	OUTPP149
GO TO 516	OUTPP150

```

416 WRITE (6,417) AU OUTPP151
417 FORMAT (32X,68M)THE PRESSURE IS CALCULATED BY INTERPOLATION FROM THOUTPP152
    *E FOLLOWING TABLE//36X,7MSTATION,14X,11MTIME (SEC),15X,11MPRESSUREOUTPP153
    *E (,2A4,1H)) OUTPP154
516 DO 17 I=1,NTB OUTPP155
    17 WRITE (6,616) I,XTB(I),ATB(I) OUTPP156
616 FORMAT (38X,12,14X,1PE12.5,15X,E12.5) OUTPP157
    GO TO 21 OUTPP158
C OUTPP159
    18 GO TO (218,318,418,518),IPRCOD OUTPP160
C ASSIGNED VARIABLE POLYNOMIAL OUTPP161
218 WRITE (6,219) AU,CX OUTPP162
219 FORMAT (40X,52M)THE AREA IS CALCULATED FROM THE FOLLOWING POLYNOMIALOUTPP163
    *L//23X,6HAREA (,A4,A1,5H) = (,1PE12.5,9H)X**3 + (,F12.5,9H)X**2 + OUTPP164
    *(,E12.5,6H)X + (,E12.5,1H)) OUTPP165
    GO TO 21 OUTPP166
518 WRITE (6,319) AU,CX OUTPP167
519 FORMAT (38X,56M)THE PRESSURE IS CALCULATED FROM THE FOLLOWING POLYNOMIALOUTPP168
    *OMIAL//20X,10MPRESSURE (,2A4,5H) = (,1PE12.5,9H)X**3 + (,E12.5,9H)OUTPP169
    *X**2 + (,E12.5,6H)X + (,E12.5,1H)) OUTPP170
    GO TO 21 OUTPP171
418 WRITE (6,419) AU,CX OUTPP172
419 FORMAT (40X,52M)THE AREA IS CALCULATED FROM THE FOLLOWING POLYNOMIALOUTPP173
    *L//23X,6HAREA (,A4,A1,5H) = (,1PE12.5,9H)T**3 + (,F12.5,9H)T**2 + OUTPP174
    *(,E12.5,6H)T + (,E12.5,1H)) OUTPP175
    GO TO 21 OUTPP176
518 WRITE (6,519) AU,CX OUTPP177
519 FORMAT (38X,56M)THE PRESSURE IS CALCULATED FROM THE FOLLOWING POLYNOMIALOUTPP178
    *OMIAL//20X,10MPRESSURE (,2A4,5H) = (,1PE12.5,9H)T**3 + (,E12.5,9H)OUTPP179
    *T**2 + (,E12.5,6H)T + (,E12.5,1H)) OUTPP180
    GO TO 21 OUTPP181
C OUTPP182
C SPECIAL AREA FUNCTION OUTPP183
19 WRITE (6,118) LSUBM,ETA OUTPP184
118 FORMAT (41X,50M)THE AREA IS CALCULATED FROM THE FOLLOWING FUNCTION/OUTPP185
    *//46X,16M1/AREA = 1 - (X/,F10.3,4H)**/,F10.5,1H)) OUTPP186
    IF (IIPSZ EQ. 4) WRITE (6,1118) D,V,3C,BETAL OUTPP187
1118 FORMAT (/6X,20M)HYDRAULIC DIAMETER =,F8.4,3H CM,7X,23M)VISCOSITY COE OUTPP188
    *FFICIENT =,E12.4,10H CM/CM-SEC,7X,6H)ETA =,F7.4) OUTPP189
    GO TO 21 OUTPP190
C OUTPP191
C ZERO VELOCITY - ASSIGNED VARIABLE NOT REQUIRED OUTPP192
20 WRITE (6,119) OUTPP193
119 FORMAT (36X,60M)THIS IS A V=0 PROBLEM - AN ASSIGNED VARIABLE IS NOT OUTPP194
    * REQUIRED) OUTPP195
C OUTPP196
C NEGLECTED SPECIES OUTPP197
21 IF (I1,NE. 0) GO TO 22 OUTPP198
    WRITE (6,120) OUTPP199
120 FORMAT (///31X,70M)NO SPECIES WILL BE PERMANENTLY NEGLECTED FROM AL OUTPP200

```

*L ERROR CONSIDERATIONS)	OUTPP201
GO TO 228	OUTPP202
22 WRITE (6,121)	OUTPP203
121 FORMAT (///31X,69HTHE FOLLOWING SPECIES WILL BE NEGLECTED FROM ALL	OUTPP204
* ERROR CONSIDERATIONS//)	OUTPP205
K = 0	OUTPP206
DO 28 I1=1,I1	OUTPP207
I = NEGL(I1)	OUTPP208
K = K + 1	OUTPP209
IF (K .EQ. 6) K = 1	OUTPP210
GO TO (23,24,25,26,27),K	OUTPP211
23 WRITE (6,122) (SPNM(J,I),J=1,2)	OUTPP212
122 FORMAT (61X,2A4)	OUTPP213
GO TO 28	OUTPP214
24 WRITE (6,123) (SPNM(J,I),J=1,2)	OUTPP215
123 FORMAT (1H+,76X,2A4)	OUTPP216
GO TO 28	OUTPP217
25 WRITE (6,124) (SPNM(J,I),J=1,2)	OUTPP218
124 FORMAT (1H+,44X,2A4)	OUTPP219
GO TO 28	OUTPP220
26 WRITE (6,125) (SPNM(J,I),J=1,2)	OUTPP221
125 FORMAT (1H+,92X,2A4)	OUTPP222
GO TO 28	OUTPP223
27 WRITE (6,126) (SPNM(J,I),J=1,2)	OUTPP224
126 FORMAT (1H+,28X,2A4)	OUTPP225
28 CONTINUE	OUTPP226
C	OUTPP227
228 IF (RMOCN) WRITE (6,1126)	OUTPP228
1126 FORMAT (///38X,56HTHE VOLUME (DENSITY) WILL BE HELD CONSTANT FOR T	OUTPP229
*HIS CASE)	OUTPP230
IF (TCN) WRITE (6,2126)	OUTPP231
2126 FORMAT (///40X,51HTHE TEMPERATURE WILL BE HELD CONSTANT FOR THIS C	OUTPP232
*ASF)	OUTPP233
RETURN	OUTPP234
C	OUTPP235
ENTRY OUT2	OUTPP236
C	OUTPP237
C INITIAL CONDITIONS	OUTPP238
KEY=2	OUTPP239
IF (.NOT.BRIEF)WRITE(6,127)	OUTPP240
127 FORMAT (1H1,52X,26H** INITIAL CONDITIONS **//)	OUTPP241
GO TO 29	OUTPP242
C	OUTPP243
ENTRY OUT3	OUTPP244
C	OUTPP245
C GENERAL OUTPUT	OUTPP246
KEY=3	OUTPP247
IF (.NEXT.OR..NOT.BRIEF)WRITE(6,128)	OUTPP248
128 FORMAT (1H1)	OUTPP249
C	OUTPP250

29	MACH = SQRT(M2)	OUTPP251
	MAX = MAX0(LS,LR)	OUTPP252
C		OUTPP253
	TENT = 0.	OUTPP254
	CSUM = 0.	OUTPP255
	PMLNG = ALNG(P*MIXMW)	OUTPP256
C	TOTAL ENTROPY AND MASS FRACTION SUM	OUTPP257
	DO 30 I=1,IS	OUTPP258
	IF (SIGMA(I),EQ, 0.) GO TO 30	OUTPP259
	TENT = TENT + SIGMA(I)*(SR(I) - ALNG(SIGMA(I)) - PMLNG)	OUTPP260
30	CSUM = CSUM + SIGMA(I)*MW(I)	OUTPP261
	TENT = TENT*1.987165	OUTPP262
C		OUTPP263
	TXXH = 0.	OUTPP264
C	ENERGY EXCHANGE RATES	OUTPP265
	DO 31 J=1,LR	OUTPP266
	XXH(J) = XX(J)*DELH(J)	OUTPP267
31	TXXH = TXXH + XXH(J)	OUTPP268
C		OUTPP269
	IF (VERST.EQ, TIMEV) GO TO 32	OUTPP270
	TIME = DVAR	OUTPP271
	X = IVAR	OUTPP272
	GO TO 33	OUTPP273
32	TIME = IVAR	OUTPP274
	X = DVAR	OUTPP275
C		OUTPP276
33	IF(BRIEF.AND..NOT.NEXT)GO TO 40	OUTPP277
	IF(UNITD.NF,FPS)GO TO 48	OUTPP278
C		OUTPP279
C	CONVERT FROM INTERNAL (CGS) UNITS TO FPS UNITS	OUTPP280
	X = X/30.48	OUTPP281
	AREAA = AREA/929.0304	OUTPP282
	DDTM = MDOT/453.59237	OUTPP283
	PP = P*2116.2	OUTPP284
	VV = V/30.48	OUTPP285
	RHOD = RHO*62.43	OUTPP286
	TT = T*1.8	OUTPP287
C		OUTPP288
	WRITE (6,129) TIME,AREAA,X,PP,NH,VV,AVH,RHOD,(SNAM(I,JCV),I=1,2),	OUTPP289
	* TT,DDTM,TENT,FRN,MACH,KOUNT,GAMMA	OUTPP290
129	FORMAT (16X,4HTIME,1PE14.5,5H SEC,14X,4HAREA,E14.5,7H SQ FT,	OUTPP291
	* 14X,14HAXIAL POSITION,F14.5,4H FT///20X,15HFLOW PROPERTIES,45X,	OUTPP292
	* 22HINTEGRATION INDICATORS//22X,8HPPRESSURE,E22.5,30X,21HSTEPS FROM	OUTPP293
	*OM LAST PRINT,9X,14/23X,10H(LR/FT**2)/22X,8HVELOCITY,E22.5,30X,	OUTPP294
	* 17HAVERAGE STEP SIZE,0PE24.5/23X,8H(FT/SEC)/22X,7HDENSITY,1PE23.5	OUTPP295
	* ,30X,20HCONTROLLING VARIABLE,11X,244/23X,10H(LR/FT**3)/22X,11HTEMP	OUTPP296
	*ERATURE,E19.5/23X,7H(DEG R)/22X,14HMASS FLOW RATE,F16.5/23X,8H(LR/	OUTPP297
	*SEC)/22X,7HENTROPY,E23.5,30X,14HRELATIVE ERROR,0PE27.5/23X,14H(OUTPP298
	*/LR/DEG R)/22X,11HMACH NUMBER,1PE19.5,30X,20HPREDICTOR ITERATIONS,	OUTPP299
	* 11X,13//22X,5HGAMMA,E25.5)	OUTPP300

C	IF (12.EQ. 0) GO TO 34	OUTPP301
	IS = 11 + 1	OUTPP302
	DO 233 K=IS, IT	OUTPP303
	KK = NEGL(K)	OUTPP304
	ESP(1,K) = SPNM(1, KK)	OUTPP305
233	ESP(2,K) = SPNM(2, KK)	OUTPP306
	WRITE (6,130) ((ESP(1,K), I=1,2), K=IS, IT)	OUTPP307
130	FORMAT (1H+, 81X, 18HELIMITED SPECIES, 13X, 2A4, 2X, 2A4/(113X, 2A4, 2X,	OUTPP308
	* 2A4))	OUTPP309
		OUTPP310
C	34 WRITE (6,131)	OUTPP311
	131 FORMAT (/56X, 19HCHEMICAL PROPERTIES//)	OUTPP312
C		OUTPP313
	CONV = 0.02883	OUTPP314
	IF (CONC .OR. EXCHR) GO TO 36	OUTPP315
C		OUTPP316
	PRINT MASS FRACTIONS AND REACTION CONVERSION RATES	OUTPP317
C	WRITE (6,132)	OUTPP318
132	FORMAT (1X, 7HSPECIES, 4X, 13HMASS FRACTION, 3X, 13HMOLE FRACTION, 3X,	OUTPP319
	* 27HNET SPECIES PRODUCTION RATE, 5X, 8HREACTION, 3X, 28HNET REACTION	OUTPP320
	* CONVERSION RATE, 3X, 13HRATE CONSTANT)	OUTPP321
	WRITE (6,133)	OUTPP322
133	FORMAT (50X, 16H(MOLE/FT**3/SEC), 11X, 6HNUMBER, 7X, 22H(MOLE-FT**3/LB	OUTPP323
	* 2/SEC), 7X, 11H(CGS UNITS))	OUTPP324
	DO 35 J=1, IR	OUTPP325
35	PRX(J) = XX(J)	OUTPP326
	CONV = 1./62.43	OUTPP327
	GO TO 37	OUTPP328
		OUTPP329
C		OUTPP330
	36 IF (CONC .OR. (.NOT. EXCHR)) GO TO 39	OUTPP331
C		OUTPP332
	PRINT MASS FRACTIONS AND ENERGY EXCHANGE RATES	OUTPP333
C	WRITE (6,134)	OUTPP334
134	FORMAT (1X, 7HSPECIES, 4X, 13HMASS FRACTION, 3X, 13HMOLE FRACTION, 3X,	OUTPP335
	* 27HNET SPECIES PRODUCTION RATE, 5X, 8HREACTION, 5X, 24HNET ENERGY EXC	OUTPP336
	* HANGE RATE, 5X, 13HRATE CONSTANT)	OUTPP337
	WRITE (6,135)	OUTPP338
135	FORMAT (50X, 16H(MOLE/FT**3/SEC), 11X, 6HNUMBER, 8X, 21H(BTU-FT**3/LB**	OUTPP339
	* 2/SEC), 7X, 11H(CGS UNITS))	OUTPP340
C		OUTPP341
C	COMPUTE MASS FRACTIONS	OUTPP342
37	DO 38 I=1, IS	OUTPP343
38	PRC(I) = SIGMA(I)*MW(I)	OUTPP344
	GO TO 44	OUTPP345
		OUTPP346
C	39 IF ((.NOT. CONC) .OR. EXCHR) GO TO 41	OUTPP347
C		OUTPP348
	PRINT MOLAR CONCENTRATIONS AND REACTION CONVERSION RATES	OUTPP349
C	WRITE (6,136)	OUTPP350

```

136 FORMAT (1X,7HSPECIES,4X,13HCONCENTRATION,3X,13HMOLF FRACTION,3X, OUTPP351
* 27HNET SPECIES PRODUCTION RATE,5X,8HREACTION,3X,24HNET REACTION COUTPP352
*ONVERSION RATE,3X,13HRATE CONSTANT) OUTPP353
WRITE (6,137) OUTPP354
137 FORMAT (12X,13H(MOLFS/FT**3),25X,16H(MOLE/FT**3/SEC),11X,6HNUMBER, OUTPP355
* 7X,22H(MOLE-FT**3/LB**2/SEC),7X,11H(CGS UNITS)) OUTPP356
DO 40 J=1,IR OUTPP357
40 PRX(J) = XX(J) OUTPP358
CONV = 1./62.43 OUTPP359
GO TO 42 OUTPP360
C OUTPP361
C PRINT MOLAR CONCENTRATIONS AND ENERGY EXCHANGE RATES OUTPP362
41 WRITE (6,138) OUTPP363
138 FORMAT (1X,7HSPECIES,4X,13HCONCENTRATION,3X,13HMOLF FRACTION,3X, OUTPP364
* 27HNET SPECIES PRODUCTION RATE,5X,8HREACTION,5X,24HNET ENERGY EXCOUTPP365
*HANGE RATE,5X,13HRATE CONSTANT) OUTPP366
WRITE (6,139) OUTPP367
139 FORMAT (12X,13H(MOLFS/FT**3),25X,16H(MOLE/FT**3/SEC),11X,6HNUMBER, OUTPP368
* 8X,21H(BTU-FT**3/LB**2/SEC),7X,11H(CGS UNITS)) OUTPP369
C OUTPP370
C COMPUTE MOLAR CONCENTRATIONS OUTPP371
42 DO 43 I=1,IS OUTPP372
43 PRC(I) = SIGMA(I)*RHOO OUTPP373
C OUTPP374
44 DO 47 IJ=1,MAX OUTPP375
IF (IJ .GT. LS .OR. IJ .GT. LR) GO TO 45 OUTPP376
FMOL = SIGMA(IJ)*MIXMW OUTPP377
WW = W(IJ)*62.43 OUTPP378
XXX = PRX(IJ)*CONV OUTPP379
WRITE (6,140) (SPNM(I,IJ),I=1,2),PRC(IJ),FMOL,WW,I,I,XXX,RATE(IJ) OUTPP380
140 FORMAT (2X,2A4,2X,1PE12.5,4X,E12.5,11X,E12.5,16X,I2,14X,E12.5,11X, OUTPP381
* E12.5) OUTPP382
GO TO 47 OUTPP383
45 IF (IJ .GT. LS) GO TO 46 OUTPP384
FMOL = SIGMA(IJ)*MIXMW OUTPP385
WW = W(IJ)*62.43 OUTPP386
WRITE (6,141) (SPNM(I,IJ),I=1,2),PRC(IJ),FMOL,WW OUTPP387
141 FORMAT (2X,2A4,2X,1PE12.5,4X,E12.5,11X,E12.5) OUTPP388
GO TO 47 OUTPP389
46 XXX = PRX(IJ)*CONV OUTPP390
WRITE (6,142) IJ,XXX,RATE(IJ) OUTPP391
142 FORMAT (79X,I2,14X,1PE12.5,11X,E12.5) OUTPP392
47 CONTINUE OUTPP393
TXXH = TXXH*0.02883 OUTPP394
WRITE (6,143) MIXMW,TXXH,CSUM OUTPP395
143 FORMAT ( /4X,26HMIXTURE MOLECULAR WEIGHT,F13.5,5X,26HTOTAL ENERGY OUTPP396
*EXCHANGE RATE,1PE15.5,7X,17HMASS FRACTION SUM,0PF14.8) OUTPP397
WRITE (6,144) OUTPP398
144 FORMAT (49X,21H(BTU-FT**3/LB**2/SEC)) OUTPP399
GO TO 78 OUTPP400

```

C	48 IF (UNITD .NE. SI) GO TO 63	OUTPP401
		OUTPP402
C		OUTPP403
C	CONVERT FROM INTERNAL (CGS) UNITS TO SI UNITS	OUTPP404
	X = X*.01	OUTPP405
	AREAA = ARFA*.0001	OUTPP406
	DOTM = MDOT*0.001	OUTPP407
	PP = P*1.01325E+05	OUTPP408
	VV = VA*.01	OUTPP409
	RHOD = RHQ*1000.	OUTPP410
	TENT = TENT*4184.0	OUTPP411
C		OUTPP412
	WRITE (6,145) TIME,AREAA,X,PP,NH,VV,AVH,RHOD,(SNAM(I,JCV),I=1,2),	OUTPP413
	* T,DOTM,TENT,ERRN,MACH,KQUNT,GAMMA	OUTPP414
	145 FORMAT (16X,4HTIME,1PE14.5,5H SEC,10X,4HAREAA,E14.5,7H SQ M,	OUTPP415
	* 14X,14HAXIAL POSITION,E14.5,4H M ///20X,15HFLOW PROPERTIES,45X,	OUTPP416
	* 22HINTEGRATION INDICATORS//22X,8HPRESSURE,E22.5,30X,21HSTEPS FROM	OUTPP417
	*DM LAST PRINT,9X,14/23X,8H(N/M**2)/22X,8HVELOCITY,E22.5,30X,17H	OUTPP418
	*RAGE STEP SIZE,0PE24.5/23X,7H(M/SEC)/22X,7HDENSITY,1PF23.5,30X,	OUTPP419
	* 20HCONTROL LING VARIABLE,11X,2A4/23X,9H(KG/M**3)/22X,11HTEMP	OUTPP420
	*F,E19.5/23X,7H(DEG K)/22X,14HMASS FLOW RATE,E16.5/23X,8H(KG/SEC)/	OUTPP421
	* 22X,7HENTROPY,E23.5,30X,14HRELATIVE ERROR,0PE27.5/23X,16H(JOULE/K	OUTPP422
	*G/DEG K)/22X,11HMACH NUMBER,1PF19.5,30X,20HPREDICTOR ITERATIONS,	OUTPP423
	* 11X,13//22X,5HGAMMA,E25.5)	OUTPP424
C		OUTPP425
	IF (12.FQ. 0) GO TO 49	OUTPP426
	IS = I1 + 1	OUTPP427
	DO 248 K=IS,IT	OUTPP428
	KK = NEGL(K)	OUTPP429
	FSP(1,K) = SPNM(1,KK)	OUTPP430
	248 ESP(2,K) = SPNM(2,KK)	OUTPP431
	WRITE (6,130) ((ESP(1,K),I=1,2),K=IS,IT)	OUTPP432
C		OUTPP433
	49 WRITE (6,131)	OUTPP434
C		OUTPP435
	CONV = 4.1840	OUTPP436
	IF (CJNC .OR. FXCHR) GO TO 51	OUTPP437
C		OUTPP438
C	PRINT MASS FRACTIONS AND REACTION CONVERSION RATES	OUTPP439
	WRITE (6,132)	OUTPP440
	WRITE (5,146)	OUTPP441
	146 FORMAT (50X,15H(MOLE/M**3/SEC),12X,6HNUMBER,7X,21H(MOLE-M**3/KG**2	OUTPP442
	*/SEC),RX,11H(CGS UNITS))	OUTPP443
	DO 50 J=1,R	OUTPP444
	50 PRX(J) = XX(J)	OUTPP445
	CONV = 0.001	OUTPP446
	GO TO 52	OUTPP447
C		OUTPP448
	51 IF (CJNC .OR. (.NOT. EXCHR)) GO TO 54	OUTPP449
C		OUTPP450

C	PRINT MASS FRACTIONS AND ENERGY EXCHANGE RATES	OUTPP451
	WRITE (6,134)	OUTPP452
	WRITE (6,147)	OUTPP453
147	FORMAT (50X,15H(MOLE/M**3/SEC),12X,6HNUMBER,7X,22H(JOULE=M**3/KG**2/SEC),7X,11H(CGS UNITS))	OUTPP454
C		OUTPP455
C	COMPUTE MASS FRACTIONS	OUTPP456
52	DO 53 I=1,18	OUTPP457
53	PRC(I) = SIGMA(I)*MH(I)	OUTPP458
	GO TO 59	OUTPP459
C		OUTPP460
54	IF ((.NOT. CONC) .OR. EXCHR) GO TO 56	OUTPP461
C		OUTPP462
C	PRINT MOLAR CONCENTRATIONS AND REACTION CONVERSION RATES	OUTPP463
	WRITE (6,136)	OUTPP464
	WRITE (6,148)	OUTPP465
148	FORMAT (12X,12H(MOLE/M**3),26X,15H(MOLE/M**3/SEC),12X,6HNUMBER,7X,21H(MOLE=M**3/KG**2/SEC),8X,11H(CGS UNITS))	OUTPP466
	DO 55 J=1,18	OUTPP467
55	PRX(J) = XX(J)	OUTPP468
	CONV = 0.001	OUTPP469
	GO TO 57	OUTPP470
C		OUTPP471
C	PRINT MOLAR CONCENTRATIONS AND ENERGY EXCHANGE RATES	OUTPP472
56	WRITE (6,138)	OUTPP473
	WRITE (6,149)	OUTPP474
149	FORMAT (12X,12H(MOLE/M**3),26X,15H(MOLE/M**3/SEC),12X,6HNUMBER,7X,22H(JOULE=M**3/KG**2/SEC),7X,11H(CGS UNITS))	OUTPP475
C		OUTPP476
C	COMPUTE MOLAR CONCENTRATIONS	OUTPP477
57	DO 58 I=1,18	OUTPP478
58	PRC(I) = SIGMA(I)*RHOO	OUTPP479
C		OUTPP480
59	DO 62 IJ=1,MAX	OUTPP481
	IF (IJ .GT. LS .OR. IJ .GT. LR) GO TO 60	OUTPP482
	FMOL = SIGMA(IJ)*MIXMW	OUTPP483
	WW = W(IJ)*1000.	OUTPP484
	XXX = PRX(IJ)*CONV	OUTPP485
	WRITE (6,140) (SPNM(I,IJ),I=1,2),PRC(IJ),FMOL,WW,IJ,XXX,RATE(IJ)	OUTPP486
	GO TO 62	OUTPP487
60	IF (IJ .GT. LS) GO TO 61	OUTPP488
	FMOL = SIGMA(IJ)*MIXMW	OUTPP489
	WW = W(IJ)*1000.	OUTPP490
	WRITE (6,141) (SPNM(I,IJ),I=1,2),PRC(IJ),FMOL,WW	OUTPP491
	GO TO 62	OUTPP492
61	XXX = PRX(IJ)*CONV	OUTPP493
	WRITE (6,142) IJ,XXX,RATE(IJ)	OUTPP494
62	CONTINUE	OUTPP495
	TXXH = TXXH*4.1840	OUTPP496
	WRITE (6,143) MIXMW,TXXH,CSUM	OUTPP497
		OUTPP498
		OUTPP499
		OUTPP500

WRITE (6,150)	OUTPP501
150 FORMAT (48X,22H(JOULE-M**3/KG**2/SEC))	OUTPP502
GO TO 78	OUTPP503
C	OUTPP504
C PRINT OUTPUT IN INTERNAL (CGS) UNITS	OUTPP505
63 WRITE (6,151) TIME,AREA,X,P,NH,V,AVH,RHD,(SNAM(I,JCV),I=1,2),	OUTPP506
* T,MDOT,TENT,ERRN,MACH,KQUNT,GAMMA	OUTPP507
151 FORMAT (16X,4HTIME,1PE14.5,5H SEC,16X,4HAREA,1E14.5,7H SQ CM,	OUTPP508
* 14X,14HAXIAL POSITION,1E14.5,4H CM//20X,15HFLOW PROPERTIES,45X,20H	OUTPP509
* 2HINTEGRATION INDICATORS//22X,8HPRESSURE,0PE23.5,29X,21HSTEPS FROM	OUTPP510
* LAST PRINT,9X,14/23X,5H(ATM)/22X,8HVELOCITY,1E20.2,32X,17HAVERAGE	OUTPP511
* STEP SIZE,1E24.5/23X,8H(CM/SEC)/22X,7HDENSITY,1PE23.5,30X,20HCONTR	OUTPP512
* LLING VARIABLE,11X,2A4/23X,10H(GM/CM**3)/22X,11HTEMPERATURE,	OUTPP513
* 0PE17.2/23X,7H(DEG K)/22X,14HMASS FLOW RATE,1PE16.5/23X,8H(GM/SEC	OUTPP514
*)/22X,7HENTROPY,0PE23.4,30X,14HRELATIVE ERROR,1E27.5/23X,14H(CAL/GM	OUTPP515
* /DEG K)/22X,11HMACH NUMBER,1E19.4,30X,20HPREDICTOR ITERATIONS,11X,	OUTPP516
* 13//22X,5HGAMMA,1E25.4)	OUTPP517
C	OUTPP518
IF (I2.EQ.0) GO TO 64	OUTPP519
IS = 11 + 1	OUTPP520
DO 263 K=IS,IT	OUTPP521
KK = NEGL(K)	OUTPP522
FSP(1,K) = SPNM(1,KK)	OUTPP523
263 FSP(2,K) = SPNM(2,KK)	OUTPP524
WRITE (6,130) ((FSP(I,K),I=1,2),K=IS,IT)	OUTPP525
C	OUTPP526
64 WRITE (6,131)	OUTPP527
C	OUTPP528
IF (CONC .OR. EXCHR) GO TO 66	OUTPP529
C	OUTPP530
C PRINT MASS FRACTIONS AND REACTION CONVERSION RATES	OUTPP531
WRITE (6,132)	OUTPP532
WRITE (6,152)	OUTPP533
152 FORMAT (50X,16H(MOLE/CM**3/SEC),11X,6HNUMBER,7X,22H(MOLE-CM**3/GM**	OUTPP534
**2/SEC),7X,11H(CGS UNITS))	OUTPP535
DO 65 J=1,IR	OUTPP536
65 PRX(J) = XX(J)	OUTPP537
GO TO 67	OUTPP538
C	OUTPP539
66 IF (CONC .OR. (.NOT. EXCHR)) GO TO 69	OUTPP540
C	OUTPP541
C PRINT MASS FRACTIONS AND ENERGY EXCHANGE RATES	OUTPP542
WRITE (6,134)	OUTPP543
WRITE (6,153)	OUTPP544
153 FORMAT (50X,16H(MOLE/CM**3/SEC),11X,6HNUMBER,8X,21H(CAL-CM**3/GM**	OUTPP545
*2/SEC),7X,11H(CGS UNITS))	OUTPP546
C	OUTPP547
C COMPUTE MASS FRACTIONS	OUTPP548
67 DO 68 I=1,IS	OUTPP549
68 PRC(I) = SIGMA(I)*MW(I)	OUTPP550

GO TO 74	OUTPP551
C	OUTPP552
69 IF ((,NOT, CONC) ,OR, EXCHR) GO TO 71	OUTPP553
C	OUTPP554
C PRINT MOLAR CONCENTRATIONS AND REACTION CONVERSION RATES	OUTPP555
WRITE (6,136)	OUTPP556
WRITE (6,154)	OUTPP557
154 FORMAT (12X,13H(MOLES/CM**3),25X,16H(MOLE/CM**3/SEC),11X,6HNUMBER,	OUTPP558
* 7X,22H(MOLE-CM**3/GM**2/SEC),7X,11H(CGS UNITS))	OUTPP559
DO 70 J=1,LR	OUTPP560
70 PRX(J) = XX(J)	OUTPP561
GO TO 72	OUTPP562
C	OUTPP563
C PRINT MOLAR CONCENTRATIONS AND ENERGY EXCHANGE RATES	OUTPP564
71 WRITE (6,138)	OUTPP565
WRITE (6,155)	OUTPP566
155 FORMAT (12X,13H(MOLES/CM**3),25X,16H(MOLE/CM**3/SEC),11X,6HNUMBER,	OUTPP567
* 8X,21H(CAL-CM**3/GM**2/SEC),7X,11H(CGS UNITS))	OUTPP568
C	OUTPP569
C COMPUTE MOLAR CONCENTRATIONS	OUTPP570
72 DO 73 I=1,IS	OUTPP571
73 PRC(I) = SIGMA(I)*RHO	OUTPP572
C	OUTPP573
74 DO 77 IJ=1,MAX	OUTPP574
IF (IJ .GT. LS .OR. IJ .GT. LR) GO TO 75	OUTPP575
FMOL = SIGMA(IJ)*MIXMW	OUTPP576
WRITE (6,140) (SPNM(I,IJ),I=1,2),PRC(IJ),FMOL,W(IJ),IJ,PRX(IJ),	OUTPP577
* RATE(IJ)	OUTPP578
GO TO 77	OUTPP579
75 IF (IJ .GT. LS) GO TO 76	OUTPP580
FMOL = SIGMA(IJ)*MIXMW	OUTPP581
WRITE (6,141) (SPNM(I,IJ),I=1,2),PRC(IJ),FMOL,W(IJ)	OUTPP582
GO TO 77	OUTPP583
76 WRITE (6,142) IJ,PRX(IJ),RATE(IJ)	OUTPP584
77 CONTINUE	OUTPP585
WRITE (6,143) MIXMW,TXXH,CSUM	OUTPP586
WRITE (6,156)	OUTPP587
156 FORMAT (49X,21H(CAL-CM**3/GM**2/SEC))	OUTPP588
C	OUTPP589
78 WRITE (6,157)	OUTPP590
157 FORMAT (/2(4X,8HVARIALE,5X,10HDERIVATIVE,6X,9HINCREMENT,6X,14HRE	OUTPP591
*LATIVE ERROR,4X))	OUTPP592
L = LSP3/2	OUTPP593
C	OUTPP594
LP = L	OUTPP595
DO 79 I=1,I	OUTPP596
LP = LP + 1	OUTPP597
WRITE (6,158) (SNAM(K,I),K=1,2),F(I),RK(I),E(I),	OUTPP598
* (SNAM(K,LP),K=1,2),F(LP),RK(LP),E(LP)	OUTPP599
158 FORMAT (2(4X,2A4,2(4X,F12.5),5X,F12.5,5X))	OUTPP600

79	CONTINUE	OUTPP601
C		OUTPP602
	IF (2* L \leq $LSP3$) GO TO 80	OUTPP603
	WRITE (6,159) (SNAM(K, $LSP3$),K=1,2),F($LSP3$),RK($LSP3$),E($LSP3$)	OUTPP604
159	FORMAT (70X,2A4,2(4X,E12.5),5X,E12.5)	OUTPP605
C		OUTPP606
80	IF (.NOT. DEBUG) GO TO 82	OUTPP607
C		OUTPP608
	DEBUG OUTPUT (INTERNAL UNITS)	OUTPP609
	IF (VERSA .NE. AREA) GO TO 83	OUTPP610
	WRITE (6,160) AA,BB,DA,D2A,DTERM	OUTPP611
160	FORMAT (1H1,12X,2HAA,18X,2HBB,13X,15HD(AREA)/D(IVAR),8X,17HD?	OUTPP612
	*)/D(IVAR)2,8X,27H(1/AREA)*D(ARFA)/D(IVAR)-AA/8X,E12.5,8X,E12.5,9X,	OUTPP613
	* E12.5,12X,E12.5,18X,E12.5)	OUTPP614
	GO TO 84	OUTPP615
83	WRITE (6,164) AA,BB,DA,D2A,DTERM	OUTPP616
164	FORMAT (1H1,12X,2HAA,18X,2HBB,11X,19HD(PRESSURE)/D(IVAR),4X,21HD?	OUTPP617
	*PRESSURE)/D(IVAR)2,4X,32H(1/PRESSURE)*D(PRESSURE)/D(IVAR)/8X,E12.5	OUTPP618
	*,8X,E12.5,9X,E12.5,12X,E12.5,18X,E12.5)	OUTPP619
84	WRITE (6,161) (I,I=1,LR)	OUTPP620
161	FORMAT (/37X,58HOMEGA(I,J) RATE OF PRODUCTION OF SPECIES I BY RE	OUTPP621
	*ACTION J//1X,7HSPECIES,55X,8HREACTION/(18X,I2,7I15))	OUTPP622
	DO 81 I=1,IS	OUTPP623
	WRITE (6,162) (SPNM(K,I),K=1,2),(OMEGA(I,J),J=1,LR)	OUTPP624
162	FORMAT (/2X,2A4,8E15.5/(10X,8E15.5))	OUTPP625
81	CONTINUE	OUTPP626
C		OUTPP627
82	IF (ABS(1.-CSUM) .LE. .001) RETURN	OUTPP628
	WRITE (6,163)	OUTPP629
163	FORMAT (7H0(OUTP),5X,19HINVALID COMPOSITION)	OUTPP630
	NEXT = .TRUE.	OUTPP631
	RETURN	OUTPP632
C		OUTPP633
C	OPTION ABBREVIATED OUTPUT DIVERTED TO POSTPROCESSOR	OUTPP634
C		OUTPP635
90	WRITE(3)KEY,TIME,AREA,X,P,V,RHO,T,MDO,TENT,MACH,GAMMA,MIXM,	OUTPP636
	SIGMA,DU41	OUTPP637
	WRITE(6,165)TIME,AREA,X,NH,AVH	OUTPP638
165	FORMAT(7H0TIME=,1PF13.6,15H SEC AREA=,E13.6,11H SQ CM,	OUTPP639
	3HX=,E13.6,19H CM NO. STEPS=,14,21H AVG. STEP SIZE=,	OUTPP640
	E13.6)	OUTPP641
	CALL CLOCK(TWALL,TPROC,MONTH,IDAY)	OUTPP642
	IF(TPROC.LT.(36.*TIMLMT))GO TO 82	OUTPP643
	WRITE(6,166)	OUTPP644
166	FORMAT(40H0(OUTP) STOP = EXCESS PROCESSOR TIME)	OUTPP645
	STOP	OUTPP646
	END	

CPADD	ADDITION OF MESH POINTS AT JET EDGE	PADD0001
	SUBROUTINE PADD(YM,NPD,REG)	PADD0002
	LOGICAL FIRST	PADD0003
	LOGICAL MCHANG	PADD0004
	INTEGER REG	PADD0005
	COMMON /UMFSH/ MCHANG,CK,DY1,NM9H , CXPC,CXTP,NRED	PADD0006
	COMMON/ INPJFT/ DUMI(12), NJ , DIMK(208)	PADD0007
	COMMON /PRNPJT/ DUMP(9),REFL,DUMPI(1003)	PADD0008
	COMMON /CTRL / DUMSX(9),C6,DUMSX1(809)	PADD0009
	DIMENSION YM(1)	PADD0010
	DATA FIRST /T/	PADD0011
C*		PADD0012
C***	REG=1-- USE CONSTANT MESH SIZE (POTENTIAL CORE)	PADD0013
C**	REG=2,3-- USE MESH DEFINED BY /UMESH/	PADD0014
C*		PADD0015
	IF(.NOT. MCHANG) GO TO 10	PADD0016
1	GO TO (10,20,20),REG	PADD0017
10	YK=YM(NPD)	PADD0018
	DY = .01*YK	PADD0019
11	NPD=NPD+1	PADD0020
	IF(.NOT.FIRST) DY=.1*CXPC*REFL/C6	PADD0021
	FIRST = .FALSE.	PADD0022
	YM(NPD)=YK+DY	PADD0023
	GO TO 100	PADD0024
C*		PADD0025
20	DYF=YM(2)-YM(1)	PADD0026
	NEW=NPD+1	PADD0027
	YM(NEW)=DYF/(CK-1.)*(CK**FLOAT(NPD)-1.)	PADD0028
	NPD=NEW	PADD0029
C*		PADD0030
100	RETURN	PADD0031
	END	PADD0032

CPBOLIC	COMPUTES SOLUTION OF PARABOLIC PDE	PBOLIC01
*		PBOLIC02
	SUBROUTINE PROLIC (K)	PBOLIC03
*		PBOLIC04
C	-J- IS THE CROSS STREAM INDEX	PBOLIC05
C	-K- IS THE INDEX FOR DEPENDENT VARIABLES	PBOLIC06
C	THETA IS THE CRANK-NICOLSON FACTOR	PBOLIC07
C	THETA=0-EXPLICIT, THETA=1-IMPLICIT, THETA=1/2, CRANK-NICOLSON	PBOLIC08
*		PBOLIC09
	COMMON /CTHETA/ THETA, II	PBOLIC10
	COMMON /CENDS/ JSTART, JEND	PBOLIC11
	COMMON /CPROLI/ G(50,1), ALPHA(1,1), BETA(50,1), GAMM(50,1),	PBOLIC12
	1 DELTA(50,1)	PBOLIC13
	COMMON /CTRIDI/ CDEFL(50), CDEFC(50), CDEFR(50), RHS(50)	PBOLIC14
	COMMON /CBNDRY/ DX, DXPRN, JT	PBOLIC15
	COMMON /CINIT/ XX, DPSI(1)	PBOLIC16
C		PBOLIC17
	NJM1=JEND-1	PBOLIC18
100	JSPI=JSTART+1	PBOLIC19
	THETA1=1.0-THETA	PBOLIC20
C*	SETUP RHS AND THEN UP-DATE BETA, GAMM, DELTA FOR NEW X-STEP	PBOLIC21
	AAA=ALPHA(II,K)*DX/(DPSI(JT)*DPSI(JT))	PBOLIC22
	AAAA=AAA*THETA1*.50	PBOLIC23
	DO 200 J=JSPI, NJM1	PBOLIC24
	J1=J	PBOLIC25
	CDEFL(J1)=AAAA*(BETA(J,K)+BETA(J-1,K))	PBOLIC26
	CDEFR(J1)=AAAA*(BETA(J,K)+BETA(J+1,K))	PBOLIC27
200	RHS(J1)=CDEFL(J1)*G(J-1,K)+(1.-CDEFL(J1)-CDEFR(J1))*(THETA1*DX*	PBOLIC28
	1 DELTA(J,K))*G(J,K)+CDEFR(J1)*G(J+1,K)+THETA1*GAMM(J,K)*DX	PBOLIC29
C*	UPDATE COEFFICIENTS ==BETA, GAMM, DELTA==	PBOLIC30
260	CALL COEFF(K)	PBOLIC31
	AAA=ALPHA(II,K)*DX/(DPSI(JT)*DPSI(JT))	PBOLIC32
	AAAA=THETA*AAA*.5	PBOLIC33
C*	RHS USES GAMM AT BOTH OLD AND NEW STATIONS	PBOLIC34
	DO 300 J=JSPI, NJM1	PBOLIC35
	J1=J	PBOLIC36
	RHS(J1)=RHS(J1)+THETA*DX*GAMM(J,K)	PBOLIC37
	CDEFR(J1)=AAAA*(BETA(J,K)+BETA(J+1,K))	PBOLIC38
	CDEFL(J1)=AAAA*(BETA(J,K)+BETA(J-1,K))	PBOLIC39
300	CDEFC(J1)=1.-CDEFL(J1)-CDEFR(J1)-THETA*DX*DELTA(J,K)	PBOLIC40
C*	AXIS OF SYMMETRY B.C.	PBOLIC41
310	J1=JSPI	PBOLIC42
	CDEFR(J1)=CDEFR(J1)-.33333333*CDEFL(J1)	PBOLIC43
	CDEFC(J1)=CDEFC(J1)+1.33333333*CDEFL(J1)	PBOLIC44
C*	OUTER (FREE-STREAM) B.C.	PBOLIC45
360	RHS(NJM1)=RHS(NJM1)-CDEFR(NJM1)*G(JEND,K)	PBOLIC46
	CALL TRIDIA(JSPI, NJM1)	PBOLIC47
390	DO 400 J=JSPI, NJM1	PBOLIC48
	J1=J	PBOLIC49
	G(J1,K)=RHS(J)	PBOLIC50

IF (G(J1,K).LT.(1.E-8)) G(J1,K)=0.
400 CONTINUE
G(1,K)=(4.*G(2,K)-G(3,K))*'.33333333
IF (G(1,K).LT.(1.E-8)) G(1,K)=0.
RETURN
END

PBOLIC51
PBOLIC52
PBOLIC53
PBOLIC54
PBOLIC55
PBOLIC56

CPARDD	COMPUTE ALL MIXED PARTIAL DERIVATIVES	PARDD001
	SUBROUTINE PARD	PARDD002
C		PARDD003
C	COMPUTE ALL MIXED PARTIAL DERIVATIVES	PARDD004
C		PARDD005
	LOGICAL TCON,RHOCON	PARDD006
C		PARDD007
	REAL LKEQ,MM,N,M,MIXMW,M2	PARDD008
C		PARDD009
	DIMENSION PXXRHO(30),PXXT(30),PXXSIG(30,25),PGSIG(25),PM2SIG(25),	PARDD010
	* PSISIG(25),PS2SIG(25),PAASIG(25),PBRSIG(25)	PARDD011
C		PARDD012
	COMMON/DPTS/VERSI,TIMEV,VERSA,AREAV,FLIM,TCON,RHOCON,IPTCOD	PARDD013
	COMMON/COND/DVAR,AREA,MODT,P,IVAR,V,RHO,T,SIGMA(25),LS,LSP3,NEXT	PARDD014
	COMMON/SPEC/SNAM(2,30),MW(25),W(25),STOIC(25,30),OMEGA(25,30)	PARDD015
	COMMON/RFAC/LSR(4,30),XX(30),RATE(30),LKEQ(30),DLKEQ(30),MM(30),LR	PARDD016
	COMMON/RRAT/A(30),N(30),EACT(30),B(30),M(25,30),ALIM1	PARDD017
	COMMON/GHSC/GRT(25),HRT(25),SR(25),CPR(25),DCPR(25)	PARDD018
	COMMON/NECC/RR,MIXMW,M2,GAMMA,TCPR,R	PARDD019
	COMMON/SABS/S1,AA,BB,S2,DA,D2A,DTERM,IRHO	PARDD020
	COMMON/DERN/F(28),ALPHA(28),BETA(28,28)	PARDD021
C		PARDD022
	DO 1 I=1,LSP3	PARDD023
	DO 1 K=1,LSP3	PARDD024
	1 BETA(I,K) = 0.	PARDD025
C		PARDD026
	DO 2 J=1,LR	PARDD027
	DO 2 I=1,LS	PARDD028
	2 PXXSIG(J,I) = 0.	PARDD029
C		PARDD030
C	XX(J) WRT RHO,T,SIGMA(I)	PARDD031
	DO 9 J=1,LR	PARDD032
	N1 = LSR(1,J)	PARDD033
	N2 = LSR(2,J)	PARDD034
	N3 = LSR(3,J)	PARDD035
	N4 = LSR(4,J)	PARDD036
	EXP1 = RATE(J)	PARDD037
	EXP2 = 1.	PARDD038
	IF (LKEQ(J) .GT. 0.) GO TO 3	PARDD039
	FXP2 = EXP(-LKEQ(J)/2.)	PARDD040
	GO TO 4	PARDD041
	3 EXP1 = EXP(ALOG(RATE(J)) - LKEQ(J))	PARDD042
C		PARDD043
	4 IF (N1 .GT. 0) GO TO 6	PARDD044
	IF (N1 .LT. 0) GO TO 5	PARDD045
	PXXRHO(J) = -MM(J)*EXP2*SIGMA(N3)*EXP1*SIGMA(N4)*FXP2	PARDD046
	PXXT(J) = MM(J)*EXP1*RHO*SIGMA(N3)*EXP2*SIGMA(N4)*EXP2*DLKEQ(J)	PARDD047
	PXXSIG(J,N2) = RATE(J)*MM(J) + XX(J)/MM(J)*M(N2,J)	PARDD048
	PXXSIG(J,N3) = -MM(J)*EXP1*RHO*EXP2*SIGMA(N4)*EXP2 + XX(J)/MM(J)*	PARDD049
	* M(N3,J)	PARDD050

PXXSIG(J,N4) = -MM(J)*EXP1*RHO*EXP2*SIGMA(N3)*EXP2	PARDD051
IF(N4.EQ.N3)PXXSIG(J,N4)=2.*PXXSIG(J,N4)	PARDD052
PXXSIG(J,N4)=PXXSIG(J,N4)+XX(J)/MM(J)*M(N4,J)	PARDD053
GO TO 9	PARDD054
5 PXXRHO(J) = -RATE(J)/RHO*SIGMA(N2)/RHO	PARDD055
PXXT(J) = FXP1*SIGMA(N3)*EXP2*SIGMA(N4)*EXP2*DLKEQ(J)	PARDD056
PXXSIG(J,N2) = RATE(J)/RHO	PARDD057
PXXSIG(J,N3) = -EXP2*(EXP1*SIGMA(N4))*EXP2	PARDD058
PXXSIG(J,N4) = -EXP2*(EXP1*SIGMA(N3))*EXP2	PARDD059
IF(N4.EQ.N3)PXXSIG(J,N4)=2.*PXXSIG(J,N4)	PARDD060
GO TO 9	PARDD061
C	PARDD062
6 IF (N4.GT. 0) GO TO 8	PARDD063
IF (N4.LT. 0) GO TO 7	PARDD064
PXXRHO(J) = MM(J)*SIGMA(N1)*RATE(J)*SIGMA(N2)	PARDD065
PXXT(J) = FXP2*MM(J)*EXP1*SIGMA(N3)*EXP2*DLKEQ(J)	PARDD066
PXXSIG(J,N1) = MM(J)*RHO*RATE(J)*SIGMA(N2) + XX(J)/MM(J)*M(N1,J)	PARDD067
PXXSIG(J,N2) = MM(J)*RHO*RATE(J)*SIGMA(N1)	PARDD068
IF(N2.EQ.N1)PXXSIG(J,N2)=2.*PXXSIG(J,N2)	PARDD069
PXXSIG(J,N2)=PXXSIG(J,N2)+XX(J)/MM(J)*M(N2,J)	PARDD070
PXXSIG(J,N3) = -EXP1*(EXP2*MM(J))*EXP2 + XX(J)/MM(J)*M(N3,J)	PARDD071
GO TO 9	PARDD072
7 PXXRHO(J) = EXP1*SIGMA(N3)*EXP2/RHO*EXP2/RHO	PARDD073
PXXT(J) = FXP1*SIGMA(N3)*EXP2/RHO*EXP2*DLKEQ(J)	PARDD074
PXXSIG(J,N1) = RATE(J)*SIGMA(N2)	PARDD075
PXXSIG(J,N2) = RATE(J)*SIGMA(N1)	PARDD076
IF(N2.EQ.N1)PXXSIG(J,N2)=2.*PXXSIG(J,N2)	PARDD077
PXXSIG(J,N3) = -EXP2*EXP1*(EXP2/RHO)	PARDD078
GO TO 9	PARDD079
C	PARDD080
8 PXXRHO(J) = 0.	PARDD081
PXXT(J) = FXP1*SIGMA(N3)*EXP2*SIGMA(N4)*EXP2*DLKEQ(J)	PARDD082
PXXSIG(J,N1) = RATE(J)*SIGMA(N2)	PARDD083
PXXSIG(J,N2) = RATE(J)*SIGMA(N1)	PARDD084
IF(N2.EQ.N1)PXXSIG(J,N2)=2.*PXXSIG(J,N2)	PARDD085
PXXSIG(J,N3) = -EXP2*(EXP1*SIGMA(N4))*EXP2	PARDD086
PXXSIG(J,N4) = -EXP2*(EXP1*SIGMA(N3))*EXP2	PARDD087
IF(N4.EQ.N3)PXXSIG(J,N4)=2.*PXXSIG(J,N4)	PARDD088
C	PARDD089
9 PXXT(J) = PXXT(J) + XX(J)*(N(J) + B(J)/T)/T	PARDD090
C	PARDD091
GTGM1 = GAMMA*(GAMMA - 1.)	PARDD092
PGAMT = 0.	PARDD093
C GAMMA WRT SIGMA(I) AND MACH NUMBER SQUARED WRT SIGMA(I)	PARDD094
DO 10 I=1,13	PARDD095
PGSIG(I) = GTGM1*(MIXW - CPR(I)/TCPR)	PARDD096
PM2SIG(I) = -M2*(MIXW + PGSIG(I)/GAMMA)	PARDD097
10 PGAMT = PGAMT + SIGMA(I)*DCPR(I)	PARDD098
C	PARDD099
C GAMMA WRT T	PARDD100

	PGAMT = -GTGM1/ICPR*PGAMT	PARDD101
C		PARDD102
C	MACH NUMBFR SQUARED WRT V	PARDD103
	PM2V = 2.*V*MIXMW/(GAMMA*R*T)	PARDD104
C	MACH NUMBFR SQUARED WRT T	PARDD105
	PM2T = -M2*(1./T + PGAMT/GAMMA)	PARDD106
C		PARDD107
	TERM = RHO	PARDD108
	IF (VERST.EQ. TIMEV) GO TO 12	PARDD109
	TERM = RHO/V	PARDD110
C	DSIGMA/DIVAR WRT V	PARDD111
	DO 11 II=4,LSP3	PARDD112
	11 BETA(II,1) = -F(II)/V	PARDD113
C		PARDD114
C	DSIGMA/DIVAR WRT RHO AND DSIGMA/DIVAR WRT T	PARDD115
	12 DO 14 II=4,LSP3	PARDD116
	I = II - 3	PARDD117
	DO 13 J=1,I,R	PARDD118
	BETA(II,2) = BETA(II,2) + STOIC(I,J)*PXXRHO(J)	PARDD119
13	BETA(II,3) = BETA(II,3) + STOIC(I,J)*PXXT(J)	PARDD120
	BETA(II,2) = F(II)/RHO + TERM*BETA(II,2)	PARDD121
14	BETA(II,3) = TERM*BETA(II,3)	PARDD122
C		PARDD123
C	DSIGMA(I)/DIVAR WRT SIGMA(K)	PARDD124
	DO 16 II=4,LSP3	PARDD125
	I = II - 3	PARDD126
	DO 16 KK=4,LSP3	PARDD127
	K = KK - 3	PARDD128
	DO 15 J=1,I,R	PARDD129
15	BETA(II,KK) = BETA(II,KK) + STOIC(I,J)*PXXSIG(J,K)	PARDD130
16	BETA(II,KK) = TERM*BETA(II,KK)	PARDD131
C		PARDD132
C	S1 WRT V,RHO,T,SIGMA(I) AND S2 WRT V,RHO,T,SIGMA(I)	PARDD133
	PS1V = 0.	PARDD134
	PS1RHO = 0.	PARDD135
	PS1T = 0.	PARDD136
	PS2V = 0.	PARDD137
	PS2RHO = 0.	PARDD138
	PS2T = 0.	PARDD139
	DO 18 II=4,LSP3	PARDD140
	I = II - 3	PARDD141
	PS1V = PS1V + BETA(II,1)	PARDD142
	PS1RHO = PS1RHO + BETA(II,2)	PARDD143
	PS1T = PS1T + BETA(II,3)	PARDD144
	PS2V = PS2V + HRT(I)*BETA(II,1)	PARDD145
	PS2RHO = PS2RHO + HRT(I)*BETA(II,2)	PARDD146
	PS2T = PS2T + HRT(I)*BETA(II,3) + CPR(I)*F(II)/T	PARDD147
	PS1SIG(I) = 0.	PARDD148
	PS2SIG(I) = 0.	PARDD149
	DO 17 KK=4,LSP3	PARDD150

AD-A045 627

GENERAL ELECTRIC CO CINCINNATI OHIO AIRCRAFT ENGINE GROUP F/G 21/5
DEVELOPMENT OF EMISSIONS MEASUREMENT TECHNIQUES FOR AFTERBURNIN--ETC(U)
OCT 75 W C COLLEY, D R FERGUSON, M A SMITH F33615-73-C-2047
R75AEG459 AFAPL-TR-75-52-SUPPL-2 NL

UNCLASSIFIED

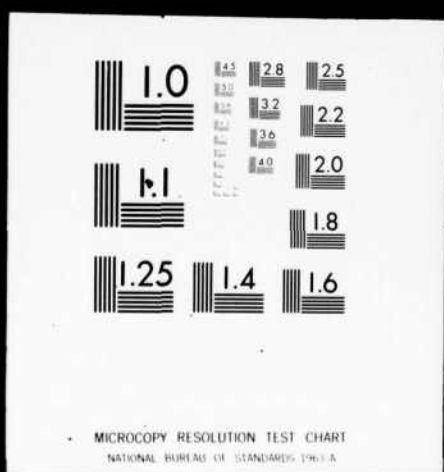
5 of 5
AD-A045 627



END
DATE
FILMED
11-77
DDC

5 OF 5

ADA045 627



K = KK = 3	PARDD151
PS1SIG(I) = PS1SIG(I) + BETA(KK,II)	PARDD152
17 PS2SIG(I) = PS2SIG(I) + HRT(K)*BETA(KK,II)	PARDD153
PS1SIG(I) = MIXMW*(PS1SIG(I) - S1)	PARDD154
18 PS2SIG(I) = MIXMW*(PS2SIG(I) - S2)	PARDD155
PS1V = MIXMW*PS1V	PARDD156
PS1RHO = MIXMW*PS1RHO	PARDD157
PS1T = MIXMW*PS1T	PARDD158
PS2V = MIXMW*PS2V	PARDD159
PS2RHO = MIXMW*PS2RHO	PARDD160
PS2T = MIXMW*PS2T - S2/T	PARDD161
C	PARDD162
GM1DG = (GAMMA - 1.)/GAMMA	PARDD163
S2DG2 = S2/(GAMMA*GAMMA)	PARDD164
C BB WRT V	PARDD165
PBBV = GM1DG*PS2V	PARDD166
C BB WRT RHO	PARDD167
PBBRHO = GM1DG*PS2RHO	PARDD168
C BB WRT T	PARDD169
PBBT = GM1DG*PS2T + S2DG2*PGAMT	PARDD170
C	PARDD171
AA WRT V	PARDD172
PAAV = PS1V - PBBV	PARDD173
C AA WRT RHO	PARDD174
PAARHO = PS1RHO - PBBRHO	PARDD175
C AA WRT T	PARDD176
PAAI = PS1T - PBBT	PARDD177
C	PARDD178
BB WRT SIGMA(I) AND AA WRT SIGMA(I)	PARDD179
DO 19 I=1,19	PARDD180
PBBSIG(I) = GM1DG*PS2SIG(I) + S2DG2*PGSIG(I)	PARDD181
19 PAA SIG(I) = PS1SIG(I) - PBBSIG(I)	PARDD182
C	PARDD183
IF (VERSA .NE. AREA) GO TO 24	PARDD184
C ASSIGNED AREA EQUATIONS	PARDD185
T1 = 1./(M2 - 1.)	PARDD186
GAM1 = GAMMA - 1.	PARDD187
C	PARDD188
DV/DIVAR WRT V	PARDD189
BETA(1,1) = T1*(DTERM - F(1)*PM2V - V*PAAV)	PARDD190
C DV/DIVAR WRT RHO	PARDD191
BETA(1,2) = -V*T1*PAARHO	PARDD192
C DV/DIVAR WRT T	PARDD193
BETA(1,3) = -T1*(V*PAAI + F(1)*PM2T)	PARDD194
C DV/DIVAR WRT SIGMA(I)	PARDD195
DO 20 II=4,LSP3	PARDD196
I = II - 3	PARDD197
20 BETA(1,II) = -T1*(V*PAA SIG(I) + F(1)*PM2SIG(I))	PARDD198
C	PARDD199
IF (RHOCNV) GO TO 22	PARDD200

C	DRHO/DIVAR WRT V	PARDD201
	BETA(2,1) = RHO*T1*(PAAV + T1*DTERM*PM2V)	PARDD202
C	DRHO/DIVAR WRT RHO	PARDD203
	BETA(2,2) = RHO*T1*PAARHO + F(2)/RHO	PARDD204
C	DRHO/DIVAR WRT T	PARDD205
	BETA(2,3) = RHO*T1*(PAAT + T1*DTERM*PM2T)	PARDD206
C	DRHO/DIVAR WRT SIGMA(I)	PARDD207
	DO 21 II=4,LSP3	PARDD208
	I = II - 3	PARDD209
	21 BETA(2,II) = RHO*T1*(PAASIG(I) + T1*DTERM*PM2SIG(I))	PARDD210
C		PARDD211
	22 IF (TCON) RETURN	PARDD212
C	DT/DIVAR WRT V	PARDD213
	BETA(3,1) = T*(GAM1*T1*(M2*PAAV + T1*DTERM*PM2V) - PBBV)	PARDD214
C	DT/DIVAR WRT RHO	PARDD215
	BETA(3,2) = T*(GAM1*M2*T1*PAARHO - PBRHO)	PARDD216
C	DT/DIVAR WRT T	PARDD217
	BETA(3,3) = T*(T1*(GAM1*(M2*PAAT + T1*DTERM*PM2T) - M2*DTERM*PGAMT	PARDD218
	*) - PRBT) + F(3)/T	PARDD219
	IF (IRHO.FQ.2) BETA(3,3) = BETA(3,3) + T*GAM1*PAAT + T*AA*PGAMT	PARDD220
C	DT/DIVAR WRT SIGMA(I)	PARDD221
	DO 23 II=4,LSP3	PARDD222
	I = II - 3	PARDD223
	BETA(3,II) = T*(T1*(GAM1*(M2*PAASIG(I) + T1*DTERM*PM2SIG(I)) - M2*	PARDD224
	* DTERM*PGSIG(I)) - PBRSIG(I))	PARDD225
	IF (IRHO.FQ.2) BETA(3,II) = BETA(3,II) + GAM1*T*PAASIG(I) + T*AA*PGSIG(I)	PARDD226
	23 CONTINUE	PARDD227
C		PARDD228
	RETURN	PARDD229
C	ASSIGNED PRESSURE EQUATIONS	PARDD230
	24 T1 = 1./((GAMMA*GAMMA)	PARDD231
C		PARDD232
C	DV/DIVAR WRT V	PARDD233
	IF (V.NE.0.) BETA(1,1) = -F(1)/V	PARDD234
C	DV/DIVAR WRT RHO	PARDD235
	BETA(1,2) = -F(1)/RHO	PARDD236
C	DV/DIVAR WRT T	PARDD237
	BETA(1,3) = 0.	PARDD238
C	DV/DIVAR WRT SIGMA(I)	PARDD239
	DO 25 II=4,LSP3	PARDD240
	25 BETA(1,II) = 0.	PARDD241
C		PARDD242
	IF (RHOCON) GO TO 27	PARDD243
C	DRHO/DIVAR WRT V	PARDD244
	BETA(2,1) = -RHO*PAAV	PARDD245
C	DRHO/DIVAR WRT RHO	PARDD246
	BETA(2,2) = F(2)/RHO - RHO*PAARHO	PARDD247
C	DRHO/DIVAR WRT T	PARDD248
	BETA(2,3) = -RHO*(PAAT + T1*DTERM*PGAMT)	PARDD249
C	DRHO/DIVAR WRT SIGMA(I)	PARDD250

DD 26 II=4, LSP3	PARDD251
I = II - 3	PARDD252
26 BETA(2,II) = -RHO*(PAASIG(I) + II*DIERM*PGSIG(I))	PARDD253
C	PARDD254
27 IF (TCON) RETURN	PARDD255
C DT/DIVAR WRT V	PARDD256
BETA(3,1) = -T*PBBV	PARDD257
C DT/DIVAR WRT RHO	PARDD258
BETA(3,2) = -T*PBBRHO	PARDD259
C DT/DIVAR WRT T	PARDD260
BETA(3,3) = BR - T*(PBBT - II*DIERM*PGANT) + F(3)/T	PARDD261
C DT/DIVAR WRT SIGMA(I)	PARDD262
DD 28 II=4, LSP3	PARDD263
I = II - 3	PARDD264
28 BETA(3,II) = -T*(PBBSIG(I) - II*DIERM*PGSIG(I))	PARDD265
C	PARDD266
RETURN	PARDD267
END	PARDD268

CPERRR	PREDICTS ERROR TO BE EXPECTED	PERRR001
	FUNCTION PFRR (H)	PERRR002
C		PERRR003
C	THIS ROUTINE PREDICTS THE ERROR WHICH CAN BE EXPECTED FROM A	PERRR004
C	STEP OF SIZE H	PERRR005
C		PERRR006
	DOUBLE PRECISION SUM	PERRR007
C		PERRR008
	DIMENSION Y(28),RK(28),E(28)	PERRR009
C		PERRR010
	COMMON/COND/DUM(5),YN(28),LS,LSP3,NEXT	PERRR011
	COMMON/SINT/HMIN,HNM1,HN,HNP1,HMAX,NH,AVH,EMAX,ERRN,JCV,KOUNT,ERRP	PERRR012
	COMMON/DERN/E(28),ALPHA(28),BETA(28,28)	PERRR013
	COMMON/PDRF/PK(28),QK(28),DUM1(56)	PERRR014
	DATA JC/0/	PERRR015
	C = H/(2.*H + HN)	PERRR016
	C0 = H/2.	PERRR017
	C1 = H/HN	PERRR018
	C2 = H + HN	PERRR019
	PERR = 0.	PERRR020
C		PERRR021
	DO 2 I=1,LSP3	PERRR022
	RK(I) = C0*(QK(I)/HN + F(I))	PERRR023
	2 E(I) = C*(C1*QK(I) + C2*(F(I) + ALPHA(I)*H))	PERRR024
C		PERRR025
	C0 = C*C2	PERRR026
	DO 4 I=1,LSP3	PERRR027
	SUM = 0.00	PERRR028
	DO 3 J=1,LSP3	PERRR029
	IF (J.EQ. I) GO TO 3	PERRR030
	SUM = SUM + BETA(I,J)*RK(J)	PERRR031
	3 CONTINUE	PERRR032
	RK(I) = (E(I) + C0*SUM)/(1. - C0*BETA(I,I))	PERRR033
	4 CONTINUE	PERRR034
C		PERRR035
	DO 6 I=1,LSP3	PERRR036
	SUM = 0.00	PERRR037
	DO 5 J=1,LSP3	PERRR038
	5 SUM = SUM + BETA(I,J)*RK(J)	PERRR039
	RK(I) = F(I) + C0*SUM	PERRR040
	6 Y(I) = YN(I) + RK(I)	PERRR041
C		PERRR042
	PERR = ERRORCY,Y,RK,E,JC,H)	PERRR043
C		PERRR044
	RETURN	PERRR045
	END	PERRR046

CPRAT IMPACT/STATIC PRESSURE RATIO PERFECT GAS

C

FUNCTION PRAT(MACH,GAMMA)
REAL MACH,MSQ

C

G=GAMMA
GP1=G+1.0
GM1=G-1.0
GEXP=G/GM1

C

MSQ=MACH**2
IF(MSQ.GT.1.0)GO TO 10

C

PRAT=(1.0+0.5*GM1*MSQ)**GEXP
RETURN

C

10 DENOM=(2.0*G/GP1+MSQ*GM1/GP1)**(GEXP/G)
PRAT=(0.5*GP1*MSQ)**GEXP/DENUM
RETURN
END

PRAT0001

PRAT0002

PRAT0003

PRAT0004

PRAT0005

PRAT0006

PRAT0007

PRAT0008

PRAT0009

PRAT0010

PRAT0011

PRAT0012

PRAT0013

PRAT0014

PRAT0015

PRAT0016

PRAT0017

PRAT0018

PRAT0019

PRAT0020

CPREDD	PERFORM ALL NECESSARY PRE-DERIVATIVE CALCULATIONS	PRED0001
	SUBROUTINE PRED	PRED0002
C		PRFDD003
C	PERFORM ALL NECESSARY PRE-DERIVATIVE CALCULATIONS	PRED0004
C		PRFDD005
	DOUBLE PRECISION DP1,DP2	PRED0006
C		PRED0007
	LOGICAL ALIM1,TCON,NEXT	PRED0008
C		PRED0009
	REAL IVAR,MDDT,LKEQ,MM,N,M,MIXMW,M2	PRED0010
C		PRED0011
	COMMON/OPIS/VERI,TIMEV,VERSA,AREAV,FLIM,TCON,RHOCNN,IPRCOD	PRED0012
	COMMON/CEND/DVAR,AREA,MDDT,P,IVAR,V,RHO,T,STGMA(25),I,S,LSP3,NEXT	PRED0013
	COMMON/SPEC/STAM(2,30),MW(25),W(25),STOIC(25,30),OMEGA(25,30)	PRED0014
	COMMON/RFAC/LSR(4,30),XX(30),RATE(30),LKEQ(30),DLKEQ(30),MM(30),LRPRED0015	
	COMMON/RRAT/A(30),N(30),EACT(30),B(30),M(25,30),ALIM1	PRED0016
	COMMON/GHSC/GRT(25),HRT(25),SR(25),CPR(25),DCPR(25)	PRED0017
	COMMON/NECC/RR,MIXMW,M2,GAMMA,TCPR,R	PRED0018
	COMMON/SABS/S1,AA,BB,S2,DA,D2A,DTERM,IRHO	PRED0019
C		PRED0020
	IF (TCON) GO TO 5	PRED0021
	GO TO 1	PRED0022
C		PRED0023
	ENTRY PRED1	PRED0024
	MWARN = 0	PRED0025
C	THERMODYNAMIC PROPERTIES	PRED0026
	1 CALL THRM (T,1)	PRED0027
C		PRED0028
	ALNGRT = A LOG(RR*T)	PRED0029
	DO 4 J=1,LR	PRED0030
C		PRED0031
C	REACTION RATE CONSTANT	PRED0032
	RATE(J) = A(J)*T**N(J)*EXP(-B(J)/T)	PRED0033
C		PRED0034
C	LN KEQ AND D(LN KEQ)/DT	PRED0035
	N1 = LSR(1,J)	PRED0036
	N2 = LSR(2,J)	PRED0037
	N3 = LSR(3,J)	PRED0038
	N4 = LSR(4,J)	PRED0039
	DELG = GRT(N3) - GRT(N2)	PRED0040
	DELH = HRT(N3) - HRT(N2)	PRED0041
	IF (N1 .GT. 0) GO TO 2	PRED0042
	DELG = DELG + GRT(N4)	PRED0043
	DELH = DELH + HRT(N4)	PRED0044
	LKEQ(J) = -DELG - ALNGRT	PRED0045
	DLKEQ(J) = (DELH - 1.)/T	PRED0046
	GO TO 4	PRED0047
2	IF (N4 .GT. 0) GO TO 3	PRFDD048
	DELG = DELG - GRT(N1)	PRED0049
	DELH = DELH - HRT(N1)	PRED0050

LKEQ(J) = -DELG + ALOGRT	PREDD051
DLKEQ(J) = (DELM + 1.)/T	PREDD052
GO TO 4	PREDD053
3 DELG = DELG + GRT(N4) - GRT(N1)	PREDD054
DELM = DELM + HRT(N4) - HRT(N1)	PREDD055
LKEQ(J) = -DELG	PREDD056
DLKEQ(J) = DELM/T	PREDD057
4 CONTINUE	PREDD058
C	PREDD059
C MIXTURE MOLECULAR WEIGHT	PREDD060
5 SSUM = 0.	PREDD061
DO 6 I=1,LS	PREDD062
6 SSUM = SSUM + SIGMA(I)	PREDD063
MIXMW = 1./SSUM	PREDD064
C	PREDD065
C ASSIGNED VARIABLE	PREDD066
IF (IPRCOD .GT. 2) GO TO 7	PREDD067
X = IVAR	PREDD068
IF (VERST .EQ. TIMEV) X = DVAR	PREDD069
CALL CINF (X,AVAR,DA,D2A)	PREDD070
GO TO 8	PREDD071
7 TIME = DVAR	PREDD072
IF (VERST .EQ. TIMEV) TIME = IVAR	PREDD073
CALL CINF (TIME,AVAR,DA,D2A)	PREDD074
C	PREDD075
C CALCULATED VARIABLE	PREDD076
8 IF (VERSA .EQ. AREAV) GO TO 9	PREDD077
P = AVAR	PREDD078
IF (V .NE. 0.) AREA = MDOT/(RHO*V)	PREDD079
GO TO 10	PREDD080
9 AREA = AVAR	PREDD081
P = RHO*RR*T/MIXMW	PREDD082
C MASS FLOW RATE	PREDD083
MDOT = RHO*AREA*V	PREDD084
C	PREDD085
10 DO 20 J=1,LR	PREDD086
N1 = LSR(1,J)	PREDD087
N2 = LSR(2,J)	PREDD088
N3 = LSR(3,J)	PREDD089
N4 = LSR(4,J)	PREDD090
C	PREDD091
C THIRD BODY FACTOR	PREDD092
MM(J) = 0.	PREDD093
IF (N1 .NE. 0 .AND. N4 .NE. 0) GO TO 13	PREDD094
IF (ALLM) GO TO 12	PREDD095
DO 11 I=1,LS	PREDD096
11 MM(J) = MM(J) + M(I,J)*SIGMA(I)	PREDD097
GO TO 13	PREDD098
12 MM(J) = SSUM	PREDD099
C	PREDD100

13	EXP1 = RATE(J)	PREDD101
	EXP2 = 1.	PREDD102
	IF (LKEQ(J) .GT. 0.) GO TO 14	PREDD103
	EXP2 = EXP(-LKEQ(J)/2.)	PREDD104
	GO TO 15	PREDD105
14	EXP1 = EXP(ALOG(RATE(J)) - LKEQ(J))	PREDD106
C		PREDD107
C	NET REACTION CONVERSION RATE	PREDD108
15	IF (N1 .GT. 0) GO TO 17	PREDD109
	IF (N1 .LT. 0) GO TO 16	PREDD110
	DP1 = RATE(J)*SIGMA(N2)	PREDD111
	DP2 = RHO*FXP1*SIGMA(N3)*EXP2*SIGMA(N4)*EXP2	PREDD112
	XX(J) = MM(J)*(DP1 - DP2)	PREDD113
	GO TO 20	PREDD114
16	DP1 = RATE(J)*SIGMA(N2)/RHO	PREDD115
	DP2 = EXP1*SIGMA(N3)*EXP2*SIGMA(N4)*FXP2	PREDD116
	XX(J) = DP1 - DP2	PREDD117
	GO TO 20	PREDD118
17	IF (N4 .GT. 0) GO TO 19	PREDD119
	IF (N4 .LT. 0) GO TO 18	PREDD120
	DP1 = RHO*SIGMA(N1)*RATE(J)*SIGMA(N2)	PREDD121
	DP2 = EXP1*EXP2*SIGMA(N3)*FXP2	PREDD122
	XX(J) = MM(J)*(DP1 - DP2)	PREDD123
	GO TO 20	PREDD124
18	DP1 = SIGMA(N1)*RATE(J)*SIGMA(N2)	PREDD125
	DP2 = EXP1*SIGMA(N3)*EXP2/RHO*EXP2	PREDD126
	XX(J) = DP1 - DP2	PREDD127
	GO TO 20	PREDD128
19	DP1 = SIGMA(N1)*RATE(J)*SIGMA(N2)	PREDD129
	DP2 = EXP2*SIGMA(N3)*EXP1*SIGMA(N4)*FXP2	PREDD130
	XX(J) = DP1 - DP2	PREDD131
20	CONTINUE	PREDD132
C		PREDD133
	RHO2 = RHO*RHO	PREDD134
	TCPR = 0.	PREDD135
	DO 22 I=1,13	PREDD136
C		PREDD137
C	TOTAL CP/R	PREDD138
	TCPR = TCPR + CPR(I)*SIGMA(I)	PREDD139
C		PREDD140
C	NET SPECIES PRODUCTION RATE	PREDD141
	W(I) = 0.	PREDD142
	DO 21 J=1,LR	PREDD143
	OMEGA(I,J) = RHO2*STOIC(I,J)*XX(J)	PREDD144
21	W(I) = W(I) + OMEGA(I,J)	PREDD145
22	CONTINUE	PREDD146
C		PREDD147
C	GAMMA (FRQZFN)	PREDD148
	GAMMA = TCPR/(TCPR - 1./MIXMW)	PREDD149
C		PREDD150

C	MACH NUMBER SQUARED	PREDD151
	M2 = V/R*V/T*MIXMW/GAMMA	PREDD152
C		PREDD153
	IF (VERSA .NE. AREAV .OR. (M2 .LT. 0.9025 .OR. M2 .GT. 1.1025))	PREDD154
	* GO TO 23	PREDD155
	MMACH = SQRT(M2)	PREDD156
	WRITE (6,101) MMACH	PREDD157
101	FORMAT (7H0(PRED),5X,7HWARNING,3X,13HMMACH NUMBER =,F8.4,19H IS APP	PREDD158
	*ROACHING 1.0)	PREDD159
	MMWARN = MMWARN + 1	PREDD160
	IF (MMWARN .LT. 15) GO TO 23	PREDD161
	WRITE (6,102)	PREDD162
102	FORMAT (7H0(PRED),5X,25H15 WARNINGS HAVE OCCURRED)	PREDD163
	NEXT = .TRUE.	PREDD164
	RETURN	PREDD165
C		PREDD166
	23 CALL DERY	PREDD167
C		PREDD168
	RETURN	PREDD169
	END	PREDD170

CPRINT	MXFLUT OUTPUT ROUTINE	PRINT001
	SUBROUTINE MPRINT	PRINT002
C		PRINT003
	COMMON /CAXIAL/ X,XL,ALOGX	PRINT004
	COMMON /CSPECI/ NSPECI,NF,DX	PRINT005
	COMMON /CREACT/ RHOREA(12)	PRINT006
	COMMON /GASIMW/ IG(2,12,2),ZMWTG(2,12,2),TAU(12,2),CPG(2,12,2)	PRINT007
	COMMON /GASCMP/ YRICH(12),YPRIME(12),FOAIR(2,12,2),ENTH(2,12,2),	PRINT008
	1CONC(16,2,12,2),DUMGAS(216)	PRINT009
	COMMON /JETDAT/ NTUBES,YREACT(12),TS(12),UREACT(12),SPV(12),	PRINT010
	1ZMWT(12),CP(12),FSPECI(12), G(12),TKE(12),OTHER(36)	PRINT011
	COMMON /CMASS/ ZMASS(12),ZMASSH(12),7MASDU(24)	PRINT012
	COMMON /CNWNET/ FDUM(100),FUELR(12), FDUMH(100)	PRINT013
C		PRINT014
	DATA JNEW,JRICH,JLEAN /2,1,2/	PRINT015
C		PRINT016
	WRITE (6,1) X	PRINT017
	WRITE (6,2)	PRINT018
	WRITE (6,3)	PRINT019
	WRITE (6,5)	PRINT020
	WRITE (6,4) (YREACT(1),FUELR(1),ZMASSH(1),RHOREA(1),UREACT(1),	PRINT021
	1YRICH(1),FOAIR(JLEAN,1,JNEW),FOAIR(JRICH,1,JNEW),G(1),TAU(1,2),	PRINT022
	2I=1,NSPECI)	PRINT023
C		PRINT024
	RETURN	PRINT025
	1 FORMAT (10X,19HAXIAL DISTANCE, X =,F13,3)	PRINT026
	2 FORMAT (//,10X,6HRAIDUS,5X,4HFUEL,8X,4HMASS,5X,7HDENSITY,2X,8HVELU	PRINT027
	1CITY,5X,4HMASS,3X,8HFUEL/GAS,2X,8HFUEL/GAS,7X,1HG,5X,7HTRANSIT)	PRINT028
	3 FORMAT (12X,5H(Y),6X,4HFLOW,8X,4HFLOW,25X,8HFRACTION,3X,4HLEAN,6X	PRINT029
	1,4HRIICH,17X,4HTIME)	PRINT030
	4 FORMAT(F20.5,3F10.5,1F10.4,5F10.5)	PRINT031
	5 FORMAT(64X,6H(RICH),33X,5H(SEC))	PRINT032
	END	PRINT033

CPRPJ	PROPERTIES OF A LAMINAR OR TURBULENT JET	PROPJ001
	SUBROUTINE PROPJ(TWO,NTURB,NREG,X,Y,T,TKE,J1,J2)	PROPJ002
	COMMON /XPRIN/ DPRIN	PROPJ003
	LOGICAL DPRIN	PROPJ004
	INTEGER TWO	PROPJ005
	LOGICAL NTURB	PROPJ006
	REAL MACH,MUL,MUEFF,KCP,MUREP	PROPJ007
C*		PROPJ008
C*		PROPJ009
C*	TWO =1 SINGLE JET	PROPJ010
C*	TWO =2 COANNULAR/COPLANAR JET	PROPJ011
C*		PROPJ012
C*	NTURB=F LAMINAR PROPERTIES ONLY	PROPJ013
C*	=T LAMINAR AND TURBULENT PROPERTIES	PROPJ014
C*	NREG =1 MIXING REGION (X,LT,XLC)	PROPJ015
C*	NREG =2 TRANSITION REGION (X,GE,XLC)	PROPJ016
C*	NREG =3 (LARGE X----- SIMILAR PROFILES)	PROPJ017
C*	X = AXIAL CO-ORDINATE (X/DJET)	PROPJ018
C*	Y = NORMAL CO-ORDINATE, FT.	PROPJ019
C*	T = TEMPERATURE , DEG R	PROPJ020
C*	TKE = TURBULENT KE, BTU/LBM	PROPJ021
C*		PROPJ022
C*	XLC = AXIAL CO-ORDINATE--START OF TRANSITION REGION	PROPJ023
C*	MACH = JET DISCHARGE MACH NUMBER	PROPJ024
C*		PROPJ025
C*		PROPJ026
	COMMON/PROPJT/	PROPJ027
	* P , PRL , PRT , RGAS , SC ,	PROPJ028
	* TREF , MUREP , MACH , XLC ,	PROPJ029
	* REFL , C , CHI , RNDM ,	PROPJ030
	* RND(200) , MUL(200) , KCP(200) ,	PROPJ031
	* MUEFF(200) , XLN(200) , DK(200) , RETURB(200)	PROPJ032
	COMMON/CPRNP/CT1,CT2,CT3,CT4,CT5,CT6,CT7,CT8,CT9,CT10	PROPJ033
	COMMON /CPRNP2/ CTP , CTS , CTM	PROPJ034
	COMMON /PROPJ2/ MACHD,REFLO,YI,YO,MERGE	PROPJ035
	LOGICAL MERGE	PROPJ036
	REAL MACHD	PROPJ037
	COMMON /MISC/ PM(10) , DUM33	PROPJ038
	LOGICAL LTERP	PROPJ039
	COMMON /LLTERP/ LTERP	PROPJ040
	COMMON /CTRL/ DUMCL1(9) , C6 , DUMCL2(809)	PROPJ041
	COMMON /EDGE / YJETE,SFEDGE	PROPJ042
	COMMON /CRMOD / CJRMOD	PROPJ043
C*		PROPJ044
	DIMENSION Y(1) , T(1) , TKE(1)	PROPJ045
	DATA GCJ/25039.7372/	PROPJ046
C*		PROPJ047
C*	MUL =LAMINAR VISCOSITY,LBM/FT*SEC	PROPJ048
C*	MUEFF =EFFECTIVE VISCOSITY,LBM/FT*SEC	PROPJ049
C*	RETURB=TURBULENCE REYNOLDS NUMBER	PROPJ050

C* KCP =KEFF/CP,LBM/FT-SEC	PROPJ051
C* XLN =TURBULENCE LENGTH SCALE,PT	PROPJ052
C* RHO =DENSITY,LBM/FT3	PROPJ053
C* DK =DIFFUSION PARAMETER FOR TURBULENCE	PROPJ054
C* PR =LAMINAR PRANDTL NUMBER	PROPJ055
C* P =PRESSURE,PSI	PROPJ056
C* SC =SUTHERLAND CONSTANT,DEG R	PROPJ057
C* TREF = REFERENCE TEMPERATURE,DEG R	PROPJ058
C* MUREF = REFERENCE VISCOSITY,LBM/FT-SEC	PROPJ059
C*	PROPJ060
C* RNORM =NORMALIZING TURBULENCE REYNOLDS NUMBER	PROPJ061
C* REFL =REFERENCE DIMENSION FOR TURB. LENGTH SCALE,PT	PROPJ062
C* PRT TURBULENT PRANDTL NUMBER	PROPJ063
C*	PROPJ064
C*	PROPJ065
C* CONSTANT VALUES	PROPJ066
C*	PROPJ067
OPR=1./PRL	PROPJ068
OPRT=1./PR	PROPJ069
CPRESS=144.*P/RGAS	PROPJ070
IF(NTURB) OREFL=1./REFL	PROPJ071
ORNORM=1./RNORM	PROPJ072
IF(TREF.NE.0.) OTREF=1./TREF	PROPJ073
CKEL=1./1.8	PROPJ074
CM=1./(1.+CT5*MACH)	PROPJ075
GO TO (300,310), TWO	PROPJ076
310 CM0=1./(1.+CT5*MACH0)	PROPJ077
CT10=2.*CT9=1.	PROPJ078
300 IF(NREG.NE.2) GO TO 33	PROPJ079
XRAT=X/XLC	PROPJ080
CN=CT4*(CT6+CT7*MACH)	PROPJ081
TRANS=XRAT**CN	PROPJ082
IF(.NOT.LTFRP) GO TO 3	PROPJ083
CM=1.0/(1.0+CT2*MACH)	PROPJ084
TEST1=CT8*(XRAT=1.)+CT1*CM*(2.-XRAT)	PROPJ085
IF(XRAT .GE. 2.) NREG=3	PROPJ086
GO TO 4	PROPJ087
3 CONTINUE	PROPJ088
TEST1=CT3*CM*TRANS	PROPJ089
IF(TEST1 .GE. CT8) NREG=3	PROPJ090
C	PROPJ091
C** MODIFICATIONS FOR SCALE CHANGE IN CORE REGION	PROPJ092
C	PROPJ093
33 YC = YJFTE*REFL/C6	PROPJ094
YINLT4= .9*YC	PROPJ095
YOUTL4= YC+.1*REFL/C6	PROPJ096
C*	PROPJ097
4 DO 100, J=J1,J2	PROPJ098
TT=T(J)	PROPJ099
RHO(J)=CPRESS/TT	PROPJ100

VISR=0.	PROPJ101
IF(IREF.NE.0.) GO TO 5	PROPJ102
C* VISCOSITY OF AIR	PROPJ103
C*	PROPJ104
MUL(J)=.98242E-6*SQRT(CKEL*YT)/(1.+261.6/YT)	PROPJ105
GO TO 10	PROPJ106
C*	PROPJ107
C* LAMINAR VISCOSITY--GENERAL	PROPJ108
C*	PROPJ109
5 TR=YT*OTREF	PROPJ110
MUL(J)=MUREF*TR*SQRT(YR)*(TREF+SC)/(YT+SC)	PROPJ111
10 IF(.NOT. NTURB) GO TO 80	PROPJ112
C*	PROPJ113
C* CALCULATION OF TURBULENT PARAMETERS	PROPJ114
C*	PROPJ115
YT=Y(J)	PROPJ116
C*	PROPJ117
C* TURBULENCE SCALE CALCULATION-- CHECK FOR TYPE OF JET	PROPJ118
C*	PROPJ119
C*	PROPJ120
C***** COMPUTATION OF LENGTH SCALE FOR TURBULENCE *****	PROPJ121
C*	PROPJ122
GO TO (20,400), TWO	PROPJ123
20 GO TO (25,35,40),NREG	PROPJ124
C*	PROPJ125
C* MIXING LAYER--REFL=WIDTH OF MIXING ZONE	PROPJ126
C*	PROPJ127
25 IF(YT,LT,YINLIM) CSCALE=C6*CJRMOD	PROPJ128
IF(YT,GE,YOULIM) CSCALE=REFL	PROPJ129
IF((YT,GE,YINLIM) .AND. (YT,LT,YOULIM)) CSCALE=C6-(C6-REFL)*	PROPJ130
* (YT-YINLIM)/(YOULIM-YINLIM)	PROPJ131
XLN(J)= CT1*CM*CSCALE	PROPJ132
GO TO 50	PROPJ133
C*	PROPJ134
C* TRANSITION REGION	PROPJ135
C*	PROPJ136
35 XLN(J)=TESTL*REFL	PROPJ137
GO TO 50	PROPJ138
C*	PROPJ139
C* SPALDING-- SEQUENCE FOR LENGTH SCALE	PROPJ140
C*	PROPJ141
40 XLN(J)=CT8*REFL	PROPJ142
GO TO 50	PROPJ143
C*	PROPJ144
C* COPLANAR/COPLANAR JET--SET LUS INDICATOR FOR UP/DN-STREAM	PROPJ145
C* OF MERGE POINT	PROPJ146
C*	PROPJ147
400 LUS=1	PROPJ148
IF(MERGE) LUS=2	PROPJ149
	PROPJ150

CPSEQ2	PSEUDO-EQUILIBRIUM COMPOSITION CH(N)-AIR-H2O SYSTEM	PSEQ2001
C	BRUTE=FORCE METHOD	PSEQ2002
C		PSEQ2003
C	SUBROUTINE PSEQ2(FAR,WAR,HC,T,BETA,FXCJ,FXND,ZZ)	PSEQ2004
C	LOGICAL FXCJ,FXND,SAMIX,TROUBL,NOSOLN,DONE	PSFQ2005
	REAL MF,K1,K2,K3,K4,K5	PSEQ2006
	DIMENSION Z(11),ZZ(11)	PSEQ2007
	DIMENSION R(3),A(9),DZ(3)	PSEQ2008
	DOUBLE PRECISION BETA	PSEQ2009
	COMMON /GHSC / FF(25),HH(25),SR(25),CPZ(25),DCPR(25)	PSEQ2010
C		PSEQ2011
	EQUIVALENCE(DSUMH,B(1)),(DSUMD,B(2)),(DSPM,B(3)),	PSEQ2012
	1(DZH2,DZ(1)),(DZ02,DZ(2)),(DZH2O,DZ(3))	PSEQ2013
C		PSEQ2014
C	ORDER OF SPECIES = H,O,H2,O2,OH,H2O,CO,CO2,N2,A,NO	PSEQ2015
C		PSEQ2016
C	TROUBL=.FALSE.	PSEQ2017
	SAMIX=(FAR-OLD FAR+WAR-OLD WAR+HC-OLD HC),EQ,0,0	PSFQ2018
	IF(SAMIX)GO TO 10	PSEQ2019
C		PSEQ2020
C	CALCULATE EQUIVALENCE RATIO	PSEQ2021
C		PSEQ2022
	OLD FAR=FAR	PSEQ2023
	OLD WAR=WAR	PSEQ2024
	OLD HC=HC	PSEQ2025
	MF=12.01+1.008*HC	PSFQ2026
	FARS=0.209495/28.9666*MF/(1.0+0.25*HC)	PSEQ2027
	ER=FAR/FARS	PSEQ2028
C		PSEQ2029
C	TOTAL POUND=ATOMS EACH CONSTITUENT PER POUND MIX	PSEQ2030
C		PSEQ2031
	GAR=1.0+FAR+WAR	PSEQ2032
	SUMH=(FAR*HC/MF+WAR*2.0/18.016)/GAR	PSEQ2033
	SUMC=(FAR/MF+3E-4/28.9666)/GAR	PSEQ2034
	SUMJ=(WAR/18.016+2.0*(0.209495+3E-4)/28.9666)/GAR	PSEQ2035
	SUMN=2.0*0.780881/28.9666/GAR	PSEQ2036
	SUMA=0.009324/28.9666/GAR	PSEQ2037
	CC=2.0*SUMC+0.5*SUMH-SUMO	PSEQ2038
	ZZX=0.5*(3E-4+28.9666/18.016*WAR)*SUMN/0.780881	PSEQ2039
	CCL=SUMC+0.5*SUMH-ZZX	PSEQ2040
	IF(CCL.LT.0.0)CCL=0.0	PSEQ2041
	CCR=0.5*(SUMO=SUMC-ZZX)	PSEQ2042
C		PSEQ2043
C	UNREACTED, COMPLETELY REACTED, AND SPECIFIED SPECIFIC MOLARITY 1/MWT	PSEQ2044
C		PSEQ2045
	SPMU=0.5*(SUMO+SUMC+SUMH+SUMN+ZZX)+SUMA	PSEQ2046
	SPMC=0.5*(0.5*SUMH+SUMN+SUMO)+SUMA	PSEQ2047
	IF(ER.GT.1.0)SPMC=0.5*(SUMH+SUMN)+SUMC+SUMA	PSEQ2048
	10 BETA1=100*BETA	PSEQ2049
		PSEQ2050

	SPH=BETA1*SPHU+BETA*SPHC	PSEQ2051
C		PSEQ2052
C	SET CONCENTRATIONS OF ISOLATED SPECIES, ZERO OTHERS.	PSEQ2053
C		PSEQ2054
	CALL SETM(1,0,0,Z,11)	PSEQ2055
	Z(10)=SUMA	PSEQ2056
	IF(FIXC)Z(7)=ZZ(7)/(ZZ(7)+ZZ(8))*SUMC	PSEQ2057
	IF(FIXN)Z(11)=ZZ(11)/(ZZ(11)+2.0*ZZ(9))*SUMH	PSEQ2058
	IF(T.EQ.OLDT)GO TO 20	PSEQ2059
C		PSEQ2060
C	GET EQUILIBRIUM CONSTANTS	PSEQ2061
C		PSEQ2062
	OLDT=T	PSEQ2063
	TXK = T / 1.8	PSEQ2064
	CALL THRM(TXK , 1.)	PSEQ2065
	K1=EXP(FF(3)+FF(4)-2.0*FF(5))	PSEQ2066
	K2=EXP(FF(3)+FF(4)-FF(6)-FF(2))	PSEQ2067
	K3=FF(3)+FF(5)-FF(6)-FF(1)	PSEQ2068
	IF(K3.GT.88.0)K3=88.0	PSEQ2069
	K3=EXP(K3)	PSEQ2070
	K4=EXP(FF(8)+FF(3)-FF(6)-FF(7))	PSEQ2071
	K5=EXP(FF(9)+FF(4)-2.0*FF(11))	PSEQ2072
C		PSEQ2073
C	INITIALIZE PRIMARY SPECIES	PSEQ2074
C		PSEQ2075
	20 IF(ER.GT.1.0)GO TO 30	PSEQ2076
	Z(3)=BETA1*CC-L=Z(7)	PSEQ2077
	IF(Z(3).LT.0.) Z(3)=0.	PSEQ2078
	Z(4)=0.5*(Z(3)+Z(7)-Z(11)=CC)	PSEQ2079
	GO TO 40	PSEQ2080
	30 Z(4)=BETA1*CCR=0.5*Z(11)	PSEQ2081
	IF(FIXC)GO TO 35	PSEQ2082
	AAA=K4-1.0	PSEQ2083
	CCC=CC+2.0*Z(4)+Z(11)	PSEQ2084
	BBB=-(CCC*AAA+K4*SUMC+0.5*SUMH)	PSEQ2085
	CCC=CCC*K4*SUMC	PSEQ2086
	Z(7)=0.5*(BBB+SQRT(BBB**2+4.0*AAA*CCC))/AAA	PSEQ2087
	35 Z(3)=CC+2.0*Z(4)-Z(7)+Z(11)	PSEQ2088
	40 Z(6)=SUMD+Z(7)-Z(11)=2.0*(SUMC+Z(4))	PSEQ2089
C		PSEQ2090
C	BEGIN ITERATION	PSEQ2091
C		PSEQ2092
	DTOL=1E-6	PSEQ2093
	IF(BETA1.LT.1E-6)DTOL=1E-7	PSEQ2094
	DO 100 I=1,30	PSEQ2095
C		PSEQ2096
C	RADICALS FROM EQUILIBRIA	PSEQ2097
C		PSEQ2098
	Z(5)=SQRT(K1*Z(3)*Z(4))	PSEQ2099
	Z(2)=K2*Z(3)*Z(4)/Z(6)	PSEQ2100

	Z(1)=K3*Z(3)*Z(5)/Z(6)	PSEQ2101
	IF(FIXCJ)GO TO 50	PSEQ2102
	Z(7)=K4*SUMC*Z(3)/(Z(6)+K4*Z(3))	PSEQ2103
50	Z(8)=SUMC-Z(7)	PSEQ2104
	IF(FIXNJ)GO TO 60	PSEQ2105
	Z(2)*5=0.25*K5*Z(4)	PSEQ2106
	Z(11)=SQRT(Z(2)*K5*(Z(2)*K5+2.0*SUNN))-Z(2)*K5	PSEQ2107
60	Z(9)=0.5*(SUNN-Z(11))	PSEQ2108
C		PSEQ2109
C	H ₂ O CONTINUITY DEFECTS AND MOLARITY DEFECT	PSEQ2110
C		PSEQ2111
	DSUMH=SUMH-Z(1)-Z(5)-2.0*(Z(3)+Z(6))	PSEQ2112
	DSUM0=SUM0-Z(2)-Z(5)-Z(6)-Z(7)-Z(11)-2.0*(Z(4)+Z(8))	PSEQ2113
	DSPM=SPM	PSEQ2114
	DO 70 J=1,11	PSEQ2115
70	DSPM=DSPM+Z(J)	PSEQ2116
C		PSEQ2117
C	TEST FOR CONVERGENCE	PSEQ2118
C		PSEQ2119
	DONE=(ABS(DSUMH).LT.OTOL*SUMH)	PSEQ2120
	1.AND.(ABS(DSUM0).LT.OTOL*SUM0)	PSEQ2121
	1.AND.(ABS(DSPM).LT.1E-8)	PSEQ2122
	IF(DONE) GO TO 110	PSEQ2123
C		PSEQ2124
C	PARTIAL DERIVATIVES TO DIRECT CONVERGENCE	PSEQ2125
C		PSEQ2126
	Z3=AMAX1(Z(3),1E-36)	PSEQ2127
	Z4=AMAX1(Z(4),1E-36)	PSEQ2128
	A(1)=2.0+0.5*(3.0*Z(1)+Z(5))/Z3	PSEQ2129
	A(2)=(Z(2)+0.5*Z(5))/Z3	PSEQ2130
	DCODX=K4*SUMC/(Z(6)+K4*Z(3))*2	PSEQ2131
	IF(.NOT.FIXCJ)A(2)=A(2)-DCODX*Z(6)	PSEQ2132
	A(3)=1.0+(1.5*Z(1)+Z(2)+0.5*Z(5))/Z3	PSEQ2133
	A(4)=0.5*(Z(1)+Z(5))/Z4	PSEQ2134
	A(5)=2.0+(Z(2)+0.5*Z(5))/Z4	PSEQ2135
	DNDD02=0.0	PSEQ2136
	IF(Z(4).NE.0.0)DNDD02=0.5*K5*Z(9)/(0.25*K5*Z4+Z(11))	PSEQ2137
	IF(.NOT.FIXNJ)A(5)=A(5)+DNDD02	PSEQ2138
	A(6)=1.0+(Z(2)+0.5*(Z(1)+Z(5)))/Z4	PSEQ2139
	IF(.NOT.FIXND)A(6)=A(6)+0.5*DNDD02	PSEQ2140
	A(7)=2.0-Z(1)/Z(6)	PSEQ2141
	A(8)=1.0-Z(2)/Z(6)	PSEQ2142
	IF(.NOT.FIXCJ)A(8)=A(8)+DCODX*Z(3)	PSEQ2143
	A(9)=1.0-(Z(1)+Z(2))/Z(6)	PSEQ2144
C		PSEQ2145
C	NEW ESTIMATES OF PRIMARY SPECIES	PSEQ2146
C		PSEQ2147
	CALL SOLV3(A,B,DZ,NDSOLN)	PSEQ2148
	TROUHL=TROUHL,OR,NDSOLN,OR,1.EQ,21	PSEQ2149
C		PSEQ2150

C	PROBLEM DIAGNOSTICS	PSEQ2151
C	IF(TROUBL)WRITE(6,1000)1,Z	PSEQ2152
1000	FORMAT(1X,I2,8X, 1P1,E11,4)	PSEQ2153
	IF(DONE)GO TO 110	PSEQ2154
C		PSEQ2155
	DZLIM = -.99*Z3	PSEQ2156
	IF(DZH2,LT,DZLIM)DZH2=DZLIM	PSEQ2157
	Z(3)=Z(3)+DZH2	PSEQ2158
	DZLIM = -.99*Z4	PSEQ2159
	IF(DZD2,LT,DZLIM)DZD2=DZLIM	PSEQ2160
	Z(4)=Z(4)+DZD2	PSEQ2161
	Z(6)=Z(6)+DZH2D	PSEQ2162
C		PSEQ2163
100	CONTINUE	PSEQ2164
C		PSEQ2165
C	PROBLEM DIAGNOSTICS	PSEQ2166
C		PSEQ2167
110	IF(TROUBL)WRITE(6,1001)FAR,WAR,HC,T,BETA,FXCO,FXNO	PSEQ2168
1001	FORMAT(15H0 I H,	PSEQ2169
	10X,1H0,10X,2HH2,9X,2H02,9X,2H0H,9X,3HH20,8X,	PSEQ2170
	12HCO,9X,3HC02,8X,2HN2,9X,2HAR,9X,2HNO/	PSEQ2171
	125H0PSEQ2 DIAGNOSTICS FAR=.F9,6,6H WAR=.F9,6,5H HC=.E10,3,	PSEQ2172
	14H T=.F9,2,7H BETA=.D17,10,8H FXCO=,L2,8H FXNO=,L2)	PSEQ2173
C		PSEQ2174
C	COMPOSITION RETURNED AS MOLE FRACTIONS	PSEQ2175
C		PSEQ2176
	CALL FMPYC(1,1,/SPM,Z,ZZ,11,1)	PSEQ2177
C		PSEQ2178
	RETURN	PSEQ2179
	END	PSEQ2180
		PSEQ2181

*QIREM=	QUADRATIC INTERPOLATION ROOT EVALUATION FOR FUNCTIONS WITH MAXIMUMS	'QIREM'	QIREM-01
C			QIREM-02
C			QIREM-03
C	SUBROUTINE QIREM(X,Y, XJP, QV)		QIREM-04
C	DIMENSION QV(8)		QIREM-05
C			QIREM-06
C	INPUT=		QIREM-07
C	X = ABSCISSA		QIREM-08
C	Y = ORDINATE (OR ERROR)		QIREM-09
C	XJP = X-JUMP TO BE TAKEN BEFORE ROOT/MAX IS SPANNED, THE SIGN		QIREM-10
C	A POSITIVE ERROR		QIREM-11
C	QV = STORAGE FOR EIGHT ELEMENT QIRE VECTOR		QIREM-12
C	QV(1) = CTR = 0. (FIRST ENTRY ONLY)		QIREM-13
C	YTOL = TOLERANCE ON THE ERROR		QIREM-14
C	YD = ORDINATE TO BE OBTAINED (OPTIONAL)		QIREM-15
C	DYDX = ESTIMATE OF SLOPE FOR 2ND GUESS (OPTIONAL)		QIREM-16
C	CTRMX = MAXIMUM NO. OF ITERATIONS (=25 IF NOT SPECIFIED)		QIREM-17
C			QIREM-18
C	OUTPUT=		QIREM-19
C	X = NEXT X ESTIMATE		QIREM-20
C	QV(1) = 0. IF YTOL HAS BEEN SATISFIED		QIREM-21
C	QV(5) = 0. IF MAX PT HAS BEEN FOUND WITHIN YTOL, AND ABS(E).GT.YTOL.		QIREM-22
C			QIREM-23
C	NOTES=		QIREM-24
C	C = THIRD COEFFICIENT IN THE EQUATION= $Y=A+B*X+C*X**2$		QIREM-25
C	= DIS IN QIRE NOTATION		QIREM-26
C	N1 = EXIT VALUE OF QV(5), N1=4 IF X IS THE PREDICTED MAX PT,		QIREM-27
C	N1=+5(-5) IF X IS JUST TO THE LEFT(RIGHT) OF THE PREVIOUS		QIREM-28
C	PREDICTED MAX PT, N1=6 IF X IS THE SECOND PT CLOSE TO THE		QIREM-29
C	OTHERWISE N1=N.		QIREM-30
C	M = ENTRY VALUE OF QV(5)		QIREM-31
C	SGM = SIGN OF M IF ABS(M)=5		QIREM-32
C	SDYDX = SIGN OF THE SLOPE OF THE CURVE		QIREM-33
C	XJ = JUMP TO BE TAKEN FROM LAST X		QIREM-34
C	XJA = ABSOLUTE VALUE OF MAXIMUM JUMP = ABS(XJP)		QIREM-35
C	XM = DISTANCE FROM CENTRAL PT TO MAX/MIN OF PARABOLA, =XMAX-XX(1)		QIREM-36
C	OR = DISTANCE FROM CENTRAL PT TO THE ROOT, =XROOT-XX(2)		QIREM-37
C	X1 = INPUT (OR LAST) X VALUE		QIREM-38
C			QIREM-39
C	COMMON /CQIREM/ YTOL,YD,DYDX,CTRMX		QIREM-40
C	COMMON /FRASE / R01,C,DXDY,E,I,II,IN,ISPAN,M,N,RADICL,SDYDX,SGN,		QIREM-41
C	TOP,X1,X13,X13P,XJ,XJA,XM, DX(3),DY(3),QV(10)		QIREM-42
C	DIMENSION XX(4),YY(4)		QIREM-43
C	EQUIVALENCE (CTR,QV(1)), (N1,QIND,QV(5)),		QIREM-44
C	(XX,QV(2)), (YY,QV(6))		QIREM-45
C			QIREM-46
C	INITIALIZING AND PRELIMINARY CHECKING		QIREM-47
C	IF(CTRMX.FQ.0.) CTRMX=25.		QIREM-48
C	DO 30 I=1,N		QIREM-49
C			QIREM-50

30	QV1(I)= QV(I)	QIREM=51
	E = Y-Y0	QIREM=52
	M = N1	QIREM=53
	IF(CTR.EQ.0.) M=0	QIREM=54
	SGM = 1.	QIREM=55
	IF(M.GE.0) GO TO 36	QIREM=56
	M = 5	QIREM=57
	SGM = -1.	QIREM=58
36	N = MIN0(M,3)	QIREM=59
C	SDYDX = SIGN(1.,-XJP)	QIREM=60
C	(ALTERNATE CALC TO CIRCUMVENT COMPILER ERROR)	QIREM=61
	IF(XJP) 41,42,42	QIREM=62
41	SDYDX = 1.	QIREM=63
	GO TO 43	QIREM=64
42	SDYDX = -1.	QIREM=65
43	XJA = ABS(XJP)	QIREM=66
	X1 = X	QIREM=67
	IF(M=5) 44,45,46	QIREM=68
44	IF(ABS(E).LE.YTOL) GO TO 800	QIREM=69
	IF(M.EQ.4 AND ABS(E-YY(2)).LE.YTOL) GO TO 700	QIREM=70
	IF(CTR.GE.CTRMAX) CALL ERROR1	QIREM=71
	GO TO 50	QIREM=72
46	M = 3	QIREM=73
45	XISP = XX(3)-XX(1)	QIREM=74
C		QIREM=75
C	DETERMINE INDEX FOR INSERTING CURRENT X,E INTO XX,YY TABLE WHICH IS	QIREM=76
C	ORDERED ACCORDING TO X.	QIREM=77
50	IN = 1	QIREM=78
	IF(N.EQ.0) GO TO 90	QIREM=79
60	IF(XX(IN).GT.X1) GO TO 70	QIREM=80
	IN = IN+1	QIREM=81
	IF(IN.LE.N) GO TO 60	QIREM=82
	GO TO 90	QIREM=83
C		QIREM=84
C	RELOCATE IN PREPARATION FOR INSERTING X,E	QIREM=85
70	II = N+1	QIREM=86
80	XX(II)= XX(II-1)	QIREM=87
	YY(II)= YY(II-1)	QIREM=88
	II = II-1	QIREM=89
	IF(II.NE.1N) GO TO 80	QIREM=90
C		QIREM=91
C	INSERT NEW POINT	QIREM=92
90	N = N+1	QIREM=93
	XX(IN)= X1	QIREM=94
	YY(IN)= E	QIREM=95
C		QIREM=96
C	LOCATE INTERVAL WHICH SPANS ROOT	QIREM=97
	ISPAN = 0	QIREM=98
	IF(N.EQ.1) GO TO 200	QIREM=99
	DO 110 I=2,N	QIREM=00

	IF(SDYDX*YY(I).GT.0. .AND. SDYDX*YY(I-1).LT.0.) ISPAN=I	QIREM-01
110	CONTINUE	QIREM-02
C		QIREM-03
C	REDUCE XX,YY TABLE TO THREE POINTS	QIREM-04
	IF(N.LE.3) GO TO 200	QIREM-05
	IF(ISPAN.EQ.0) GO TO 140	QIREM-06
C	(ROOT HAS BEEN SPANNED)	QIREM-07
122	IF(ISPAN.EQ.N) GO TO 150	QIREM-08
	IF(ISPAN.EQ.2) GO TO 175	QIREM-09
	IF(ABS(YY(1)).GT.ABS(YY(4))) GO TO 140	QIREM-10
	GO TO 175	QIREM-11
C		QIREM-12
C	(ROOT HAS NOT BEEN SPANNED)	QIREM-13
140	IF(IN.LE.2) GO TO 175	QIREM-14
C		QIREM-15
C	DELETE FIRST POINT	QIREM-16
150	DO 160 I=1,N	QIREM-17
	XX(I) = XX(I+1)	QIREM-18
160	YY(I) = YY(I+1)	QIREM-19
	ISPAN = ISPAN-1	QIREM-20
C	DELETE FOURTH POINT	QIREM-21
175	N = N-1	QIREM-22
C		QIREM-23
C	SIMPLE X-JUMP PREDICTION	QIREM-24
200	N1 = N	QIREM-25
	IF(ISPAN.GT.0 .OR. DYDX.NE.0.) GO TO 205	QIREM-26
C	XJ = SDYDX*SIGN(XJA,-E)	QIREM-27
C	(ALTERNATE CALC TO CIRCUMVENT COMPILER ERROR)	QIREM-28
	XJ = XJP	QIREM-29
	IF(E.LT.0.) XJ=-XJ	QIREM-30
	GO TO 900	QIREM-31
C		QIREM-32
C	CURVE FIT PREDICTIONS	QIREM-33
205	IF(N=2) 210,220,300	QIREM-34
C		QIREM-35
C	ONE POINT PREDICTION BASED ON INPUT VALUE OF DXDY	QIREM-36
210	XJ = -E/DYDX	QIREM-37
	GO TO 900	QIREM-38
C		QIREM-39
C	TWO POINT STRAIGHT LINE PREDICTION	QIREM-40
220	BOT = YY(2)-YY(1)	QIREM-41
	IF(BOT.EQ.0.) GO TO 230	QIREM-42
	DXDY = (XX(2)-XX(1))/BOT	QIREM-43
	IF(DXDY*SDYDX.GT.0.) GO TO 240	QIREM-44
C	(CURVE SLOPE IS WRONG - MOVE TOWARD MAXIMUM POINT)	QIREM-45
230	XJ = -3.*SDYDX*XJA	QIREM-46
	GO TO 900	QIREM-47
C	(CURVE SLOPE IS CORRECT)	QIREM-48
240	XJ = -E*DXDY	QIREM-49
	GO TO 900	QIREM-50

C		QIREM-51
C	PARABOLIC CURVE FIT PREDICTION	QIREM-52
	300 DX(1) = XX(1)-XX(2)	QIREM-53
	DX(3) = XX(3)-XX(2)	QIREM-54
	DY(1) = YY(1)-YY(2)	QIREM-55
	DY(3) = YY(3)-YY(2)	QIREM-56
	BOT = DX(1)*DY(3) - DX(3)*DY(1)	QIREM-57
	IF(ABS(BOT).LT.1.E-12) GO TO 600	QIREM-58
	TOP = DX(1)*DX(1)*DY(3) - DX(3)*DX(3)*DY(1)	QIREM-59
	XH = .5*TOP/BOT	QIREM-60
	X13 = XX(3)-XX(1)	QIREM-61
	IF(ABS(XH).GT.ABS(1.E3*X13)) GO TO 600	QIREM-62
	C = BOT/(DX(1)*DX(3)*X13)	QIREM-63
	RADICL = XH*XH - YY(2)/C	QIREM-64
	IF(RADICL.LF.0.) GO TO 360	QIREM-65
	SGN = SIGN(1.,SDYDX*C)	QIREM-66
	XH = XH + SGN*SQRT(RADICL)	QIREM-67
	GO TO 890	QIREM-68
C	(IMAGINARY ROOT, HENCE WE ARE LOOKING FOR THE MAXIMUM POINT.	QIREM-69
C	PREDICT MAX PT IF M=3, SELECT PTS ON LEFT/RIGHT SIDE OF PREVIOUSLY	QIREM-70
C	PREDICTED PT IF M=4/5)	QIREM-71
	360 IF(M=4) 363,364,365	QIREM-72
	363 IF(ABS(XH).LT.XJA) N1=4	QIREM-73
	GO TO 890	QIREM-74
	364 XJ = -X13/8.	QIREM-75
	N1 = 5	QIREM-76
	IF(IN.GT.2) GO TO 900	QIREM-77
	XJ = -XJ	QIREM-78
	N1 = -5	QIREM-79
	GO TO 900	QIREM-80
	365 XJ = SGN*X13P/4.	QIREM-81
	N1 = 6	QIREM-82
	GO TO 900	QIREM-83
C		QIREM-84
C	RETREAT TO LINEAR INTERPOLATION	QIREM-85
	600 IF(ISPAN.GT.0) GO TO 122	QIREM-86
	GO TO 140	QIREM-87
C		QIREM-88
C	MAXIMUM FOUND	QIREM-89
	700 QIND = 0.	QIREM-90
	GO TO 930	QIREM-91
C		QIREM-92
C	SOLUTION FOUND	QIREM-93
	800 CTR = 0.	QIREM-94
	GO TO 930	QIREM-95
C		QIREM-96
C	FINIS	QIREM-97
	890 X1 = XX(2)+XH	QIREM-98
	GO TO 910	QIREM-99
	900 X1 = X1+XJ	QIREM-00

```
910 CALL DTEST
      X = AMAX1(XX(1)-XJA,AMIN1(X1,XX(N)+XJA))
      CTR = CTR+1.
930 DO 950 I=1,8
950 QV(I) = QV1(I)
999 RETURN
      END
```

```
QIREM=01
QIREM=02
QIREM=03
QIREM=04
QIREM=05
QIREM=06
QIREM=07
```

CRATCON COMPUTATION OF RATE CONSTANTS

	SUBROUTINE RATCON(P)	RATCON01
	REAL K5,K6,K7,K8,K9,KR5,KR6,KR7,KR8,KR9	RATCON02
	COMMON /CKINET/ DUM(6),NREAC,	RATCON03
	* RC1(9),RC2(9),RC3(9),TBR(11,5),RCON(7)	RATCON04
	COMMON /PSFO / F0A,BFTS,TP,X1(1),X2,X3,X4,X5,X6,XK(15)	RATCON05
	COMMON /GHSC / F(25),HH(25),SR(25),CP2(25),DCPR(25)	RATCON06
	COMMON /CKH02 / K5(1),K6,K7,K8,K9	RATCON07
	DIMENSION X(6)	RATCON08
	EQUIVALENC (X(1),X1(1))	RATCON09
C		RATCON10
C	**REACTIONS** (1) H +OH +M = H2O+M	RATCON11
C	(2) H +H +M = H2 +M	RATCON12
C	(3) H +O +M = OH +M	RATCON13
C	(4) O +O +M = O2 +M	RATCON14
C	(5) H +O2 +M = HO2+M	RATCON15
C	(6) H +HO2 = OH +OH	RATCON16
C	(7) OH +HO2 = H2O+O2	RATCON17
C	(8) O +HO2 = OH +O2	RATCON18
C	(9) H +HO2 = H2 +O2	RATCON19
C		RATCON20
C		RATCON21
C	IF NREAC.LT.5, THE HYDROPEROXYL REACTIONS ARE NOT CONSIDERED	RATCON22
C		RATCON23
C	DATA C2/.01603286/	RATCON24
	TK = TP/1.8	RATCON25
	1 CALL SFTM(1,0,RCON,5)	RATCON26
	NRC = NREAC	RATCON27
	IF(NRC.EQ.5) NRC=9	RATCON28
C		RATCON29
	DO 10 L=1,NREAC	RATCON30
	TB = 0.	RATCON31
	DO 5 K=1,11	RATCON32
	5 TB = TB+TBR(K,L)*X(K)	RATCON33
	10 RCON(L)= TB*RC1(L)*TK** (RC2(L))*EXP(-RC3(L)/(1.98596*TK))	RATCON34
	20 CALL THRM(TK,1.)	RATCON35
	RHOM = 144.*P*C2/(1545.32*TP)	RATCON36
	IF(NRC.LT.5) RETURN	RATCON37
C		RATCON38
C	SECTION TO CALCULATE HYDROPEROXYL ADJUSTED RCON(5)	RATCON39
C		RATCON40
	DO 21 L=1,5	RATCON41
	K = L+4	RATCON42
	TB = 1.	RATCON43
	IF(L.NE.1) GO TO 21	RATCON44
	TB = 0.	RATCON45
	DO 22 J=1,11	RATCON46
	22 TB = TB+TBR(J,5)*X(J)	RATCON47
	21 K5(L) = TB*RC1(K)*TK** (RC2(K))*EXP(-RC3(K)/(1.98596*TK))	RATCON48
	RCON(5)= K5(1)	RATCON49
		RATCON50

IF(X1.LE.0.) GO TO 50	RATCON51
FQ6 = FXP(F(1)+F(13)-2.*F(5))	RATCON52
FQ7 = FXP(F(5)+F(13)-F(6)-F(4))	RATCON53
EQ8 = FXP(F(2)+F(13)-F(5)-F(4))	RATCON54
FQ9 = FXP(F(1)+F(13)-F(3)-F(4))	RATCON55
KR6 = K6/EQ6	RATCON56
KR7 = K7/EQ7	RATCON57
KR8 = K8/EQ8	RATCON58
KR9 = K9/EQ9	RATCON59
FQ5 = FXP(F(1)+F(4)-F(13))	RATCON60
KR5 = RHOM*K5/(P*FQ5/14.69594)	RATCON61
XNUMR= RHOM*(KR5+KR5/K5*(KR6*X5**2/(X1*X4)+KR9*X3/X1+KR7*X6/X1	RATCON62
* +KR8*X5/X1)	RATCON63
DENOM = RHOM*(KR5+(K6+K9)*X1+K7*X5+KR8*X2)	RATCON64
RCON(5)= RCON(5)*(1.-XNUMR/DENOM)	RATCON65
IF(RCON(5).LE.0.) RCON(5)=0.	RATCON66
50 RETURN	RATCON67
END	RATCON68
	RATCON69

CREADIN	INPUT ROUTINE FOR EMISSIONS PROG.	READIN01
C	SORT, CONVERT AND STORE	READIN02
C	SUBROUTINE READIN	READIN03
C	LOGICAL NODATA	READIN04
	REAL MWT, MATRX, LMV	READIN05
	DIMENSION AMBT(5), CYCLE(5), FUEL(3), AIDAT(10,11), CAROL(4,11),	READIN06
	RADII(11), STDVAL(10), TITLE2(10), ZATR(11)	READIN07
	COMMON /CBITS / BITS, BLANK	READIN08
	COMMON /TRDUHL/ ERR, ERRMAJ, INERR, PRERR	READIN09
	LOGICAL FRR, FRRMAJ, INEPR, PREPR	READIN10
C	COMMON/INDATA/HC, HF, HAR, TT2, RHETA, TT25, FFAR5, FIC, PPO,	READIN11
	RRCD(11), RRCU2(11), RRHC(11), RRNDX(11), PT(11), PS(11), BLOC(11),	READIN12
	EICOC(11)	READIN13
	COMMON/JETDAT/NPTS, RAD(12), TSJ(12), UJ(12), SPV(12), MWT(12),	READIN14
	CP(12), FRAR(12), SPLDG(12), OTHER1(12,4)	READIN15
	COMMON/GASCOMP/HICH(12,2), F(2,12,2), H(2,12,2), Z(16,2,12,2),	READIN16
	HCINCP(12,2), OTHER2(2,12,2,4)	READIN17
	COMMON/UPCTRL/TITLE(20), PRINT(30)	READIN18
C	EQUIVALENC(RAD(1), RADII(1)), (P0, AMBT(1)), (T0, AMBT(2)),	READIN19
	(HUM, AMBT(3)), (V0, AMBT(4)), (RADJ, AMBT(5)), (T2, CYCLE(1)),	READIN20
	(FAR5, CYCLE(2)), (EIND2C, CYCLE(3)), (BETA, CYCLE(4)), (T25, CYCLE(5))	READIN21
C	DATA STDVAL/14,696,518.69,0.00531,0.0,1.0,	READIN22
	518.69,0.02,20.0,0.0,518.69/	READIN23
	DATA TITLE2(1)	READIN24
	/60HCONSUMPTION OF EMISSIONS IN AUGMENTOR EXHAUST PLUME /	READIN25
C	NAMELIST/ISTDAT/AMBT,P0,T0,HUM,V0,RADJ,	READIN26
	CYCLE,T2,FAR5,EIND2C,BETA,T25,FUEL,	READIN27
	AIDAT,CAROL,RADII,PT,PS,BLOC,EICOC,	READIN28
	TITLE,SF,PRINT	READIN29
C	RESET ALL INPUT VARIABLES TO STANDARD VALUES	READIN30
C	CALL SETM(1,BLANK,TITLE,20)	READIN31
	CALL MOVE(2,STDVAL(1),AMBT,5,1,STDVAL(6),CYCLE,5,1)	READIN32
	CALL SFIM(5,BITS,CAROL,44,RADII,11,PT,44,AIDAT,110,PRINT,30)	READIN33
	FUEL(1)=2.0	READIN34
	FUEL(2)=537.0	READIN35
	FUEL(3)=BITS	READIN36
	SF=1.0	READIN37
C	READ ENGINE TEST DATA AND CYCLE PARAMETERS	READIN38
C	READ(5,ISTDAT)	READIN39
	IF(INERR) RETURN	READIN40
		READIN41
		READIN42
		READIN43
		READIN44
		READIN45
		READIN46
		READIN47
		READIN48
		READIN49
		READIN50

C		READIN51
C	MOVE GENERAL DATA INTO COMMON BLOCKS	READIN52
C	WAR=NUM	READIN53
	TT2=2	READIN54
	BETA=BETA	READIN55
	TT25=TT25	READIN56
	FFARS=FARS	READIN57
	FIC=1E-3*EINOC	READIN58
	PP0=P0	READIN59
	HC=FUEL(1)	READIN60
	LHV=FUEL(3)	READIN61
C		READIN62
C	ESTIMATE FUEL HEATING VALUE IF NOT GIVEN	READIN63
C	IF(LHV.EQ.BITS)LHV=(184686.04+37977.7*HC)/(11.91468+HC)	READIN64
	HF=LHV-1066.4605*(158.1830+47.0276*HC)/(12.01+1.008*HC)	READIN65
	DIF=FUEL(2)-537.0	READIN66
	HF=HF+DIF*(0.43733823+DIF*(3.0451344E-4+DIF*(3.7682468E-8	READIN67
	+DIF*1.2234568E-10)))	READIN68
C		READIN69
C	SUBSTITUTE CHANGE VECTORS, IF ANY, INTO ALDAT	READIN70
C	SET STANDARD VALUES IF NO VALUE GIVEN IN INPUT LIST	READIN71
C	IF(ALDAT(9,1).EQ.BITS)ALDAT(9,1)=0.0	READIN72
	DO 10 N=1,11	READIN73
	IF(CAROL(1,N).NE.BITS)CALL MOVE(1,CAROL(1,N),ALDAT(1,N),4,1)	READIN74
	IF(ALDAT(1,N).EQ.BITS)GO TO 20	READIN75
	IF(RADII(N).NE.BITS)ALDAT(5,N)=RADII(N)	READIN76
	IF(PI(N).NE.BITS)ALDAT(6,N)=PI(N)	READIN77
	IF(PS(N).NE.BITS)ALDAT(7,N)=PS(N)	READIN78
	IF(ALDAT(7,N).EQ.BITS)ALDAT(7,N)=P0	READIN79
	IF(BLOC(N).NE.BITS)ALDAT(8,N)=BLOC(N)	READIN80
	IF(ALDAT(8,N).EQ.BITS)ALDAT(8,N)=BETA	READIN81
	IF(EICOC(N).NE.BITS)ALDAT(9,N)=EICOC(N)	READIN82
	IF(ALDAT(9,N).EQ.BITS)ALDAT(9,N)=ALDAT(9,1)	READIN83
C		READIN84
C	MOVE PROBE DATA INTO COMMON BLOCKS	READIN85
C	RRCD(N)=SF*ALDAT(1,N)	READIN86
	RRCP(N)=SF*ALDAT(2,N)	READIN87
	RRHC(N)=SF*ALDAT(3,N)	READIN88
	RRNOX(N)=SF*ALDAT(4,N)	READIN89
	PI(N)=ALDAT(6,N)	READIN90
	PS(N)=ALDAT(7,N)	READIN91
	BLOC(N)=ALDAT(8,N)	READIN92
	EICOC(N)=ALDAT(9,N)*IF-3	READIN93
	RAD(N)=ALDAT(5,N)	READIN94
C		READIN95
C	SET UP HETEROGENEOUS GAS MODEL	READIN96
		READIN97
		READIN98
		READIN99
		READIN00

C		READIN01
	CALL MXFITO(N)	READIN02
	10 NPTS=N	READIN03
C		READIN04
C	SET UP AMBIENT AIR PROPERTIES	READIN05
C		READIN06
	20 NA=NPTS+1	READIN07
	RAD(NA)=RADJ	READIN08
	TSJ(NA)=T0	READIN09
	UJ(NA)=V0	READIN10
	FBAH(NA)=0.0	READIN11
	SPLDG(NA)=0.0	READIN12
	RICH(NA,1)=0.0	READIN13
	F(1,NA,1)=0.0	READIN14
	F(2,NA,1)=0.0	READIN15
	CALL EQGAST(0.0,WAR,MC,T0,P0,FALSE,FALSE,ZAIR,	READIN16
	HAIR,MWT(NA),SX,SPV(NA),AX,CP(NA))	READIN17
	CALL SETM(1,0.0,0,Z(1,1,NA,1),32)	READIN18
	CALL SETM(1,HAIR,H(1,NA,1),2)	READIN19
	CALL MOVE(2,ZAIR,Z(1,1,NA,1),1,1,ZAIR,Z(1,2,NA,1),1,1)	READIN20
C		READIN21
C	SET STANDARD PRINT STATIONS IF NOT SPECIFIED	READIN22
C		READIN23
	IF(PRINT(1).NE.BITS) GO TO 999	READIN24
	PRINT(1)=0.1*RADJ	READIN25
	PRINT(2)=10.0*RADJ	READIN26
C		READIN27
C	GENERATE INPUT FOR JETMIX	READIN28
C		READIN29
	999 CALL JETPRF	READIN30
C		READIN31
	RETURN	READIN32
	END	READIN33

CREADT	READS -JETMIX- OUTPUT TAPE	READT001
C		READT002
C*	OCTOBER 1973 MODIFICATIONS FOR AUG EMIS SYS	READT003
C		READT004
	SUBROUTINE READT (IENTRY)	READT005
C		READT006
	COMMON /CBITS / BITS,BLANK	READT007
	INTEGER BLANK	READT008
	COMMON /CJETOT/ GCJ,DIAJ,VJET,EJET,NXTA	READT009
	COMMON /CSTART/ GJET(200),GEX	READT010
	COMMON /CCOUNT/ ZX(5),NSL(5),NX,XMAX	READT011
	COMMON /CINPUT/ ZPSI(100,5),ZR(100,5),ZU(100,5),ZE(100,5),	READT012
	1 ZRHQ(100,5),ZXLN(100,5),ZF(100,5)	READT013
	COMMON /CFILK/ CSC	READT014
	COMMON /CPRFI / PSI(200),Y(200),UD(200),ED(200),RHQ(200),XLN(200),	READT015
	* ALX(100,12),NPD,FSPECI(12),NC,X(100)	READT016
	REAL MOLF1(100),MOLF2(100),MOLF3(100),MOLF4(100),MOLF5(100),	READT017
	1 MOLF6(100),MOLF7(100),MOLF8(100),MOLF9(100),MOLF10(100),	READT018
	2 MOLF11(100),MOLF12(100)	READT019
	INTEGER TEST,BOY,XXX	READT020
	LOGICAL FOUND	READT021
	EQUIVALENCE (FUEL, FSPECT)	READT022
	EQUIVALENCE (MOLF1(1),ALX(1,1)),(MOLF2(1),ALX(1,2)),(MOLF3(1),	READT023
	1 ALX(1,3)),(MOLF4(1),ALX(1,4)),(MOLF5(1),ALX(1,5)),(MOLF6(1),	READT024
	2 ALX(1,6)),(MOLF7(1),ALX(1,7)),(MOLF8(1),ALX(1,8)),(MOLF9(1),	READT025
	3 ALX(1,9)),(MOLF10(1),ALX(1,10)),(MOLF11(1),ALX(1,11)),	READT026
	4 (MOLF12(1),ALX(1,12))	READT027
C		READT028
	COMMON / KEYS / KEYA(11), KEYB(11), KODA(11), KODB(11)	READT029
C		READT030
	DIMENSION FUEL(12)	READT031
	DIMENSION DUMMY1(2000),DUMMY2(98),DUMMY3(73),DUMMY4(1311),	READT032
	* DUMMY5(59)	READT033
	DIMENSION THD(200),IID(200),U(200),I(200),IDI(200),	READT034
	* XMACH(200),PTUT(200),TID(200),PTD(200)	READT035
	DIMENSION CNAMF(12),ALE(12),SCH(12),CPC(36)	READT036
C		READT037
	DATA KPREJT/6HPREJET/	READT038
	DATA JETMIX,BOY,TEST,XXX /6HJETMIX,3HBOY,4HTEST,2HXX/	READT039
C		READT040
	NAMelist /NLPROF/ NPD,PSI,Y,UD,ED,RHQ,XLN,MOLF1,MOLF2,MOLF3,	READT041
	1 MOLF4,MOLF5,MOLF6,MOLF7,MOLF8,MOLF9,MOLF10,MOLF11,MOLF12	READT042
	NAMelist / JFTXX / GCJ, DIAJ, VJET, FJET, X, NXTA, NC	READT043
	NAMelist / REFDAT / GJET, FUEL	READT044
	NAMelist /RDATA/ DIAJ,VJET,EJET,GEX,GCJ,NXTA,NC,X	READT045
C		READT046
C*	SAVE KEYA(4)	READT047
C*		READT048
	K4SV = KEYA(4)	READT049
C		READT050

*	IF (IENTRY.GT.0) GO TO 100	READT051
*	CSC=1000.	READT052
C		READT053
C*	READ JETMIX DATA FROM TAPE	READT054
C*	KEYA(4) = JETMIX	READT055
	KEYA(7) = XXX	READT056
	READ (2) KXX1,KREC,	READT057
	* DUMMY5,BITS,ERK,GC,GCJ,	READT058
	* FOJT,DIAJ,MJET,TJET,PTJET,VJET,TIJET,EJET,PF,VE,ME,TIE,TE,	READT059
	* AXI,VJ,NY,UF,MIXPR,XLC,FL,MJ,MERGE,VV,CUN1,	READT060
	* CT1,CT2,CT3,CT4,CT5,CT6,CT7,CT8,CT9,CTP,CTS,CTM,	READT061
	* GAM,RG,PR,PRY,SC,TREF,MUREF,SP,SV,SLEN,DPRIN,PLUT,C6,	READT062
	* MIX,CF,MAXIT,TOL,SUPB,	READT063
	* X,DUMMY1,NXTA,I,NC,DUMMY2	READT064
C*		READT065
	XMAX=X(NXTA)	READT066
	NX=5	READT067
	IF (NXTA.LT.5) NX=NXTA	READT068
	NXREM=NXTA-NX	READT069
C*		READT070
C*	READ INITIAL -FAR- AND -G- PROFILES FROM PREJET DATA	READT071
C*		READT072
	KEYA(4) = KPREJT	READT073
	KEYA(7) = BLANK	READT074
	READ (1)	READT075
	* DIAJ,MJET,TIJET,VJET,PE,TE,TIE,VE,GFX,RG,PR,PRI,SC,TREF,	READT076
	* MURFF,DIFF,NC,CHAME,ALE,SCM,CPC,NJ,NM,CT1,CT2,CT3,CT4,CT5,	READT077
	* CT6,CT7,CT8,GJET,Y,VD,THD,IID,ED,	READT078
	* MOLF1,MOLF2,MOLF3,MOLF4,MOLF5,MOLF6,	READT079
	* MOLF7,MOLF8,MOLF9,MOLF10,MOLF11,MOLF12,	READT080
	* DUMMY3,FUEL	READT081
C*		READT082
C		READT083
	NREAD=1	READT084
	GO TO 110	READT085
100	IF (NXREM.LE.0) GO TO 230	READT086
	IF (NXREM.EQ.1) NX=2	READT087
	IF (NXREM.EQ.2) NX=3	READT088
	IF (NXREM.EQ.3) NX=4	READT089
	NXREM=NXREM-NX+1	READT090
	NREAD=2	READT091
*		READT092
C*	RESET KEYA(4) - - READ REMAINDER OF JETMIX DATA	READT093
C*		READT094
	110 CONTINUE	READT095
	NSPECI = NC	READT096
	KEYA(4) = JETMIX	READT097
	DO 210 I=NREAD,NX	READT098
		READT099
		READT100

IACI=I+IENTRY	READT101
ZX(I)=X(IACI)	READT102
XX=ZX(I)	READT103
KEYA(7) = CSC * XX + 0.5	READT104
FOUND=.FALSE.	READT105
READ (2) JREC,KXX,KREG,	READT106
* SUPD,SUPSTP,CORE,CORSTP,MER,MERSTP,NPD,	READT107
* PSI,Y,UO,TMO,ED,TID,RMO,XLN,U,T,TOT,XMACH,	READT108
* PTOT,TTO,PTD,	READT109
* MOLF1,MOLF2,MOLF3,MOLF4,MOLF5,MOLF6,	READT110
* MOLF7,MOLF8,MOLF9,MOLF10,MOLF11,MOLF12,J	READT111
IF (KXX,EQ,KEYA(7)) FOUND=.TRUE.	READT112
IF (.NOT.FOUND) GO TO 210	READT113
NSL(I)=NPD	READT114
DO 200 J=1,NPD	READT115
ZPSI(J,I)=PSI(J)	READT116
ZW(J,I)=Y(J)	READT117
ZU(J,I)=UO(J)	READT118
ZE(J,I)=ED(J)	READT119
ZRHJ(J,I)=RHJ(J)	READT120
ZXLN(J,I)=XLN(J)	READT121
ZF(J,I)=0.	READT122
C* CALCULATE AVERAGE FUEL/AIR RATIO	READT123
DO 190 K=1,NSPECI	READT124
ZF(J,I)=ALX(J,K)*FSPECI(K)+ZF(J,I)	READT125
190 CONTINUE	READT126
200 CONTINUE	READT127
210 CONTINUE	READT128
C*	READT129
GEX=0.	READT130
230 CONTINUE	READT131
C* RESTORE KEYA(4)	READT132
C*	READT133
KEYA(4) = K48V	READT134
C*	READT135
RETURN	READT136
END	READT137

*READT2	READS **JET4IX** TAPE (MXFLUT)	READT201
C		READT202
	SUBROUTINE READT	READT203
E		READT204
	COMMON /CFILK/ CSC	READT205
	COMMON /JMXIC/ NAME(10),TITLE1(10),IDENT(10),ADDRES(10),	READT206
	* IDENT1(10),TMODT(9),ERK,GC,GCJ,FOOT,	READT207
	* EJET,AXI,NJ,NM,UE,	READT208
	* MIXPRE,XLC,FLONJ,MERGE,NV,CON1,CT1,CT2,CT3,CT4,CT5,CT6,	READT209
	* CT7,CT8,CT9,CTP,CTG,CTH,SC,IRFF,MUREF,	READT210
	* SX,SV,SLEN,DPRIN,PLDT,C6,MIX,CF,MAXIT,TOL,SUPB,	READT211
	* XPRN(100),N(100),JC(100),TC(100),TIC(100),	READT212
	* PTC(100),WJ(100),YJ(100),TIC(100),YSDNIC(100),YCB(100),	READT213
	* XD(100),RD(100),YH(100),YCD(100),PD(100),WV(100),	READT214
	* MA2(100),VE2(100),TE2(100),NXTA,NC,CNAME(12),ALJ(12),	READT215
	* ALJO(12),ALF(12),SCM(12),CPC(36),DIEF	READT216
C		READT217
	COMMON /JMX2/ SUPD,SUPSTP,CORE,CORSTP,NER,MERSTP,NPD,	READT218
	* PSI(200),Y(200),U(200),ED(200),TID(200),RHO(200),	READT219
	* XLN(200),U(200),TDT(200),XMACH(200),PTOT(200),TID(200),	READT220
	* PTD(200),ALX(100,12),T(200)	READT221
	COMMON /JETDAT/ NTUBES,YREACT(12),TS(12),UREACT(12),SPV(12),	READT222
	IZMWT(12),CP(12),FSPECI(12),GHAT(12),TKF(12),OTHER(36)	READT223
	COMMON /GASCMP/ GASX(1104)	READT224
	COMMON /CINPUT/ XY(5),R(5,50),UU(5,50),RRHO(5,50),XSPECI(5,50,13),	READT225
	1PPSI(5,50),GG(5,50),HJJ(5),NII	READT226
	COMMON /CXIDCA/ XX(5)	READT227
	COMMON /CSPECI/ NSPECI,NE,DX	READT228
	COMMON /CINPJT/ DIAJ,MJE',TJET,PTJET,VJET,TIJET,PE,VF,ME,TIE,TE,	READT229
	IX(100),GAM,RG,PR,PRT	READT230
	COMMON /CAXIAL/ XDUM,XL,ALOGX	READT231
C		READT232
	COMMON /CBITS/ BITS,BLANK	READT233
	COMMON / KEYS / KEYA(11),KEYB(11),KODA(11),KODB(11)	READT234
	COMMON / FILES / ORGF,UPDF,NEW,SCRF	READT235
	INTEGER ORGF,UPDF,SCRF	READT236
	COMMON / STCTRL / DUMST1(3),FIRSTH,DUMST2(13)	READT237
	LOGICAL FIRSTH	READT238
	COMMON /CSPARE/ H(1600),PSIR(50),G(50),SY(50),RAD(50)	READT239
	COMMON /CPRINT/ PDUM(20)	READT240
	COMMON /XISAVE/ I,ISTAR	READT241
C		READT242
	DIMENSION SP(13),SPE(12),SPR(12)	READT243
C		READT244
	REAL MOLF1(100),MOLF2(100),MOLF3(100),MOLF4(100),MOLF5(100),	READT245
	1MOLF6(100)	READT246
	REAL MOLF7(100),MOLF8(100),MOLF9(100),MOLF10(100),	READT247
	* MOLF11(100),MOLF12(100)	READT248
	DIMENSION THD(200)	READT249
	EQUIVALENCE (MOLF1(1),ALX(1,1)),(MOLF2(1),ALX(1,2)),	READT250

*	(MOLF3(1),ALX(1,3)),(MOLF4(1),ALX(1,4)),	READT251
*	(MOLF5(1),ALX(1,5)),(MOLF6(1),ALX(1,6)),	READT252
*	(MOLF7(1),ALX(1,7)),(MOLF8(1),ALX(1,8)),	READT253
*	(MOLF9(1),ALX(1,9)),(MOLF10(1),ALX(1,10)),	READT254
*	(MOLF11(1),ALX(1,11)),(MOLF12(1),ALX(1,12))	READT255
	LOGICAL FOUND	READT256
	INTEGER TEST,BDY,XXX,BLANK	READT257
C		READT258
	DATA JETMIX,BDY,TEST,XXX /6HJETMIX,3HBDY,4HTEST,2HXX/	READT259
	DATA ISPALD /6HSPALDG/	READT260
	DATA SPE/12A-1./	READT261
C		READT262
	NAMelist /NLPREF/ UD,MOLF1,MOLF2,MOLF3,MOLF4,MOLF5,MOLF6,	READT263
*	MOLF7,MOLF8,MOLF9,MOLF10,MOLF11,MOLF12,	READT264
	1 RHD,XLN,Y,U,T,XMACH,NPD,P3I	READT265
	NAMelist /JETXX/ VJEI,X,DIAJ,NC	READT266
	NAMelist /NLSPLD/ G,SY,RAD,JEND	READT267
C		READT268
	NREACT=NTUHES	READT269
	K4SV = KEYA(4)	READT270
	IF (.NOT. FIRSTM) GO TO 90	READT271
C		READT272
	CALL REDPRF	READT273
	CALL NEWFLO	READT274
	FIRSTM= .FALSE.	READT275
C		READT276
	NJ=0	READT277
C		READT278
	CALL SETH(1,BITS,X,50)	READT279
	NPD=1	READT280
	KEYA(4) = JETMIX	READT281
	KEYA(7)= XXX	READT282
	READ (2) KXX1,KREC,	READT283
*	NAME,ITITLE1,IDEN1,ADDRES,IDEN11,TWOOT,BIIS,ERK,GC,GCJ,	READT284
*	FUDT,DIAJ,MJET,TJET,PTJFS,VJET,TIJET,EJET,PE,VE,MF,TIF,TE,	READT285
*	AXI,NJ,NM,UF,MIXPRE,XLC,FLOWJ,MERGE,WV,CUN1,	READT286
*	CT1,CT2,CT3,CT4,CT5,CT6,CT7,CT8,CT9,CTP,CTS,CTM,	READT287
*	GAM,RG,PR,PRI,SC,TREF,MUREF,SK,SV,SI EN,DPRIN,PLOT,C6,	READT288
*	MIX,CF,MAXIT,TUL,SUPR,	READT289
*	X,XPRN,B,UC,IC,IIC,PTC,WJ,YJ,IIC,YSONIC,	READT290
*	YCD,XD,RD,YR,YCD,PD,WV,MAP,VE2,TE2,NXTA,I,	READT291
*	NC,CNAME,ALJ,ALJD,ALE,SCM,CPC,DIFF,CSC	READT292
C*		READT293
	DO 80 KK=2,50	READT294
	IF(X(KK).EQ.BITS) GO TO 85	READT295
	IF(X(KK).EQ.X(KK-1)) GO TO 84	READT296
80	CONTINUE	READT297
84	IK = KK+1	READT298
	DO 86 JK=IK,50	READT299
86	X(JK-1)=X(JK)	READT200

85 CONTINUE	READT201
C* NSPECI= NC	READT202
NTUBES= NSPECI-1	READT203
NF = NSPECI+1	READT204
I=1	READT205
ISTAR=1	READT206
NII=5	READT207
90 IF (ISTAR.GT.NII) I=NII	READT208
100 XX(I)=X(ISTAR)	READT209
IF (XX(I).EQ.BITS) GO TO 210	READT210
C* HAVE REACHED END OF STORAGE ON JETMIX TAPE	READT211
C	READT212
XY(I)=ALOG(XX(I)+1.)	READT213
C* --ISTAR-- KEEPS TRACK OF CURRENT POSITION IN JETMIX X-VECTOR	READT214
C* --I -- IS THE LOCATION OF DATA IN STORAGE IN MXFLUT	READT215
KEYA(4)= JETMIX	READT216
CSC=1000.	READT217
KEYA(7) = CSC * XX(I) + 0.5	READT218
FOUND=.FALSE.	READT219
READ (2) JREC,KXX,KREG,	READT220
* SUPD,SUPSTP,CONF,CURSTP,NER,MERSTP,NPD,	READT221
* PSI,Y,UD,TMD,ED,TID,RHO,XIN,U,T,TOT,XMACH,	READT222
* PTOT,TID,PID,	READT223
* MOLF1,MOLF2,MOLF3,MOLF4,MOLF5,MOLF6,	READT224
* MOLF7,MOLF8,MOLF9,MOLF10,MOLF11,MOLF12,J	READT225
IF (KXX.EQ.KEA(7)) FOUND=.TRUE.	READT226
IF (.NOT.FOUND) GO TO 210	READT227
IF (PDUM(15).NE.0.) WRITE (6,700) X(ISTAR)	READT228
700 FORMAT (/10X,2H*,3X,13HRESTART AT X=,F12.6)	READT229
KEYA(4)= ISPALD	READT230
READ (3) KXX,SY,RAD,G,JEND	READT231
C	READT232
NJ = MIN(50,NPD)	READT233
PSIR(1)=0.	READT234
C	READT235
UCONST= .02	READT236
UEDGE = UCONST*(U(1)-U(NPD)) + U(NPD)	READT237
JINTER=NPD	READT238
136 IF (U(JINTER).GE.UEDGE) GO TO 140	READT239
JINTER=JINTER-1	READT240
IF (JINTER.EQ.2) GO TO 140	READT241
GO TO 136	READT242
140 NINTER=NPD-JINTER+1	READT243
CALL LSPFIT(U(JINTER),PSI(JINTER),NINTER,UEDGE,PSIEDG,1,0)	READT244
DPSI=(PSIEDG-PSI(1))/FLUAT(NJ-1)	READT245
IF (PDUM(15).NE.0.) WRITE (6,138) UEDGE,PSIEDG,NINTER,JINTER,	READT246
* XX(I)	READT247
138 FORMAT (1X,27HUEDGE,PSIEDG,NINTER,JINTER,1HX,2F10.6,213,F10.6)	READT248
C* RESTORE DATA ON ZERO-TO-ONE- BASIS (PERCENT JET WIDTH)	READT249
	READT250

CALL NEWPSI(PSIR,NJ)	READT251
CALL FMPYC(1,PSIEDG,PSIR,PSIR,50)	READT252
DO 151 J=1,NPD	READT253
151 Y(J) = Y(J)**2	READT254
CALL LSPFIT (PSI,Y,NPD,PSIR,H,NJ,0)	READT255
CALL LSPFIT (PSI,U,NPD,PSIR,H(NJ+1),NJ,0)	READT256
CALL LSPFIT (PSI,RHD,NPD,PSIR,H(2*NJ+1),NJ,0)	READT257
C	READT258
DO 160 K=1,VSPECI	READT259
KP2NJ1=(K+2)*NJ+1	READT260
CALL LSPFIT (PSI,ALX(1,K),NPD,PSIR,H(KP2NJ1),NJ,0)	READT261
NPDM1=NPD-1	READT262
C	READT263
C* CHECK ON JETMIX CALCULATIONS	READT264
SP(K)=0.	READT265
DO 155 J=1,NPDM1	READT266
155 SP(K)=SP(K)+(ALX(J+1,K)+ALX(J,K))*(PSI(J+1)-PSI(J))*5	READT267
160 CONTINUE	READT268
C	READT269
C* STORE **G** RESULTS FROM--SPLDG--	READT270
KP2NJ1= (VSPECI+3)*NJ+1	READT271
CALL LFIT (SY,G,JEND,PSIR,H(KP2NJ1),NJ,0)	READT272
C	READT273
SP(NF)=0.	READT274
DO 161 K=1,VSPECI	READT275
IF (SPE(K).LT.0.) SPE(K)=SP(K)	READT276
SPR(K)=SP(K)/SPE(K)	READT277
161 SP(NF)= SP(NF)+SP(K)	READT278
IF (PDUM(15).NE.0.) WRITE (6,2) SP,SPR	READT279
2 FORMAT (1X,29HSPEICE CONSERVAFIUN IN JETMIX,/,8H SPECIE,8F12.8,/,	READT280
18H RATIO,8F12.4)	READT281
PSISCL= VJET*DIAJ**2/144.*3.1415927	READT282
DIAJQ2= .5*DIAJ	READT283
NJJ(1)=NJ	READT284
DO 200 J=1,NJ	READT285
PSSI(1,J)= PSIR(J)*PSISCL	READT286
R(1,J)= DIAJQ2**2*H(J)	READT287
NJP1=NJ+J	READT288
UU(1,J)=H(NJP1)	READT289
NJX2P1=NJ*2+J	READT290
RRHJ(1,J)=H(NJX2P1)	READT291
XSPECI(1,J,NF)=0.	READT292
C	READT293
C* CALCULATE AVERAGE MIXTURE RATIO	READT294
DO 190 K=1,VSPECI	READT295
KP2NJ1=(K+2)*NJ+J	READT296
XSPECI(1,J,K)=H(KP2NJ1)	READT297
190 XSPECI(1,J,NF)=XSPECI(1,J,K)*FSPECI(K)+XSPECI(1,J,NF)	READT298
KP2NJ1= (VSPECI+3)*NJ+J	READT299
GG(1,J)=H(KP2NJ1)	READT200

200 CONTINUE

ISTAR=ISTAR+1

I=ISTAR

IF (I,LE,NII) GO TO 100

GO TO 220

C

C* NEARING END OF JETMIX RUN

210 NII=NII-1

IF (XL,LE,X(ISTAR-1)) GO TO 220

IF(PDUM(15),NE,0.) WRITE (6,1) XL,X(ISTAR-1)

XL=X(ISTAR-1)

C

220 KEVA(4) = K49V

RETURN

C

1 FORMAT (10X,20(2H*),//,10X,21HREQUESTED END VALUE OF X IS BEY

10ND FND OF JETMIX RUN, XL HAS BEEN RESET,/,20X,17HOLD VALUE OF XL

1=,F10.6,/,20X,17HNEW VALUE OF XL =,F10.6)

END

READT201

READT202

READT203

READT204

READT205

READT206

READT207

READT208

READT209

READI210

READT211

READT212

READT213

READT214

READT215

READI216

READT217

READT218

READT219

C	CHREDPRE	-REDPRE= SUB. TO READ PREJET DATA	REDPRE01
C			REDPRE02
C		NOV, 1973	REDPRE03
C			REDPRE04
		SUBROUTINE REDPRE	REDPRE05
C			REDPRE06
	COMMON /INDATA/	N, HF, HAR, T2, BETA, T25, FARS, EINO2C, PO,	REDPRE07
*		RCO(11), RCO2(11), RHC(11), RNOX(11),	REDPRE08
*		PT(11), PS(11), BLOC(11), QCO(11)	REDPRE09
	REAL	N	REDPRE10
	COMMON /JETDAT/	NPTS, RAD(12), TS(12), UJ(12), SPV(12), MNT(12),	REDPRE11
*		CP(12), FULL(12), SPALDG(12), TKE(12), OTHER1(36)	REDPRE12
	REAL	MNT	REDPRE13
	COMMON /GASCOMP /	RICH(24), FUUL(48), ENTH(48),	REDPRE14
*		CONC(768), MCINCP(24), OTHER2(192)	REDPRE15
	COMMON /DPCTRL/	TITLE(20), PRINT(30)	REDPRE16
	COMMON /GASTMW/	TG(2,12,2), DUMTMW(120)	REDPRE17
C			REDPRE18
	COMMON / KEYS /	KEYA(11), KEY4(11), KODA(11), KODH(11)	REDPRE19
	DIMENSION	GJET(200), Y(200), UD(200), THD(200), TID(200), ED(200),	REDPRE20
*		MOLF1(100), MOLF2(100), MOLF3(100), MOLF4(100),	REDPRE21
*		MOLF5(100), MOLF6(100), MOLF7(100), MOLF8(100), MOLF9(100),	REDPRE22
*		MOLF10(100), MOLF11(100), MOLF12(100)	REDPRE23
	DIMENSION	CNAME(12), ALE(12), SCH(12), CPC(36)	REDPRE24
C			REDPRE25
	DIMENSION	H(2,12,2)	REDPRE26
	EQUIVALENCE	(H(1,1,1), ENTH(1))	REDPRE27
	DATA	KPREJT/6H2PREJET/	REDPRE28
C			REDPRE29
	NAMelist / REFDAT /	NPTS, RAD, TS, U, SPV, MNT, CP.	REDPRE30
	1	FUEL, SPALDG, TKE, OTHER1,	REDPRE31
	2	RICH, FUUL, ENTH, CONC, MCINCP, OTHER2	REDPRE32
*		, TITLE, PRINT	REDPRE33
*		N, HF, HAR, T2, BETA, T25, FARS, EINO2C, PO, RCO, RCO2, RHC, RNOX,	REDPRE34
*		PT, PS, BLOC, QCO	REDPRE35
C			REDPRE36
	K4SV	= KEYA(4)	REDPRE37
	KEYA(4)	= KPREJT	REDPRE38
C			REDPRE39
	READ	(1)	REDPRE40
*		DIAJ, HJET, TIJET, VIJET, PE, TF, TIF, VE, GFX, RG, PR, PRT, SC, TREF,	REDPRE41
*		MURFF, DIFF, NC, CNAME, ALE, SCH, CPC, NJ, NM, CT1, CT2, CT3, CT4, CT5,	REDPRE42
*		CT6, CT7, CT8, GJET, Y, UD, THD, TID, ED,	REDPRE43
*		MOLF1, MOLF2, MOLF3, MOLF4, MOLF5, MOLF6,	REDPRE44
*		MOLF7, MOLF8, MOLF9, MOLF10, MOLF11, MOLF12,	REDPRE45
*		NPTS, RAD, TS, UJ, SPV, MNT, CP, FUEL, SPALDG, TKE, OTHER1,	REDPRE46
*		TITLE, PRINT, N, HF, HAR, T2, BETA, T25, FARS, EINO2C, PO,	REDPRE47
*		RCO, RCO2, RHC, RNOX, PT, PS, BLOC, QCO,	REDPRE48
*		RICH, MCINCP, FUUL, ENTH, CONC, OTHER2	REDPRE49
*		, GEX	REDPRE50

C	KEYA(4) = K4SV	REDPRE51
C		REDPRE52
C	INITIALIZE RICH FRACTION TEMPERATURE	REDPRE53
C		REDPRE54
C	DO 10 KK=1,NPTS	REDPRE55
	CALL TFMH1(CONC,H(1,KK,1),TG(1,KK,1),CP)	REDPRE56
10	CONTINUE	REDPRE57
C		REDPRE58
C	RETURN	REDPRE59
C		REDPRE60
C	END	REDPRE61
		REDPRE62

CRH2	HYDROGEN CONCENTRATION ESTIMATED FROM CO MEASUREMENT	RH200001
C	REF ITR GLEASON TO STABRYLLA 3/8/71	RH200002
C		RH200003
	FUNCTION RH2(RCQ1)	RH200004
C		RH200005
	IF(RCQ1.EQ.RCQ)GO TO 30	RH200006
C		RH200007
	RCQ=RCQ1	RH200008
	IF(RCQ.GT.0.091022292)GO TO 20	RH200009
C		RH200010
	H2CO=0.7*(100.0*RCQ)**0.16	RH200011
	GO TO 30	RH200012
C		RH200013
	20 H2CO=0.0147*(100.0*RCQ)**1.342	RH200014
C		RH200015
	30 RH2=H2CO*RCQ	RH200016
	RETURN	RH200017
	END	RH200018

CSCALE	TURBULENCE SCALE/SINGLE, COANNULAR/COPLANAR JETS	SCALE001
	SUBROUTINE SCALE(UU, THOJ, NREG, X)	SCALE002
C*****	CONTROL COMMON	SCALE003
C*		SCALE004
	COMMON /CTRL/	SCALE005
*	NXTA , CMPS , QJET , TURBJ , COEF(10)	SCALE006
*	NPU , NPD , DXC , XU , XDD	SCALE007
*	DSTOR(800)	SCALE008
C*		SCALE009
C*****	PROFILE COMMON	SCALE010
C*		SCALE011
	COMMON /PRNF/ PST(200), Y(200), UD(200), THD(200), ED(200)	SCALE012
C*****	CONSTANT AND ERROR COMMON	SCALE013
C*		SCALE014
	COMMON /CNERR/ BITS , ERR , GC , GCJ , FOOT	SCALE015
C*		SCALE016
C*****	BOUNDARY CONDITION COMMON	SCALE017
C*		SCALE018
	COMMON /BC/ UEDGE , EEDGE , YHEDGE	SCALE019
C*		SCALE020
	COMMON /BC0/ U0, E0, TH0	SCALE021
	COMMON /CTRL2/	SCALE022
*	EDGE1 , SFI , MERGE , XMERGE , YMERGE ,	SCALE023
*	SLOPE1 , SLOPE2 , CEPT1 , CEPT2	SCALE024
	LOGICAL MERGE	SCALE025
	DIMENSION UU(1)	SCALE026
	INTEGER THOJ	SCALE027
	COMMON /EDGE/ YJETE, SFEDGE	SCALE028
	COMMON /PRNPJ2/ MACHD, REFLO, YI, YO, MFRGP	SCALE029
	LOGICAL MERGP	SCALE030
	COMMON /PRNPJT/	SCALE031
*	P , PRL , PRY , RGAS , SC	SCALE032
*	TREF , MUREF , MACH , XLC	SCALE033
*	REFL , C , CHI , RNDRM	SCALE034
*	RHO(200) , MUL(200) , KCP(200)	SCALE035
*	MUEFF(200) , XLN(200) , DK(200) , RETURB(200)	SCALE036
	COMMON /CPROP/ CT1, CT2, CT3, CT4, CT5, CT6, CT7, CT8	SCALE037
C*		SCALE038
	COMMON /MIXER/ MIX, RD(100), XD(100), CF, YR(100)	SCALE039
	LOGICAL MIX	SCALE040
	COMMON /FLNBAL/ MAXIT, SUPR, NIT, PSID, YDD, YDC,	SCALE041
*	P1, P2, UCL, TOL, UPSTRM, CVG	SCALE042
	LOGICAL SUPR, CVG, UPSTRM	SCALE043
	COMMON /ACONVG/ YCD(100), PD(100), INDE(100), CHOKE, CHOKED	SCALE044
	LOGICAL CHOKE, CHOKED	SCALE045
	COMMON /DEIT/ CLSP(100)	SCALE046
	COMMON /STA2/ MACH2, TS2, SS2, V2, RHO2, NPD2	SCALE047
	REAL MACH2	SCALE048
	COMMON /BCMIX2/ GRADU, TW, MUW, RHOW, PTF, YTE	SCALE049
	REAL MUW	SCALE050

COMMON /CHNDY/	YCB(100),CLSPCB(100),YCB1 , UCL1	SCALE051
COMMON /JUTMIX/	NXORIG	SCALE052
COMMON /SCALD/	SCLD,ALXLIM	SCALE053
LOGICAL	SCLD	SCALE054
C*	EQUIVALENCF (C6,COEF(6))	SCALE055
C*	1 GO TO (100,200) , TWOJ	SCALE056
C*		SCALE057
C*	SINGLE JET== COMPUTE LOCAL WIDTH OF MIXING ZONE	SCALE058
C*		SCALE059
	100 GO TO (140,120,120) , NREQ	SCALE060
	120 REFL=C6*YJFTE	SCALE061
	GO TO 160	SCALE062
	140 IF(SCLD) GO TO 142	SCALE063
	DO 143 L=1,NPU	SCALE064
	IF(UU(L).NF,UCL1) GO TO 144	SCALE065
	143 CONTINUE	SCALE066
	GO TO 144	SCALE067
	142 DO 1442 L=1,NPU	SCALE068
	IF(UU(L),GE,ALXLIM) GO TO 144	SCALE069
	1442 CONTINUE	SCALE070
	144 REFL=C6*(YJETE-Y(L=1))	SCALE071
C*		SCALE072
	160 RETURN	SCALE073
C*		SCALE074
C*		SCALE075
C*		SCALE076
C*	COANNULAR/COPLANAR JET	SCALE077
C*		SCALE078
C*	TEST MERGE TO DETERMINE IF UP/DN-STREAM OF MERGE STATION	SCALE079
C*		SCALE080
	200 TEST1=U0+.1.E=6	SCALE081
	TEST2=U0+.1.E=6	SCALE082
	IF(MERGE) GO TO 260	SCALE083
C*		SCALE084
C*	UPSTREAM== COMPUTE REFL,REFLO,YI,YO	SCALE085
C*		SCALE086
	210 YD=C6*YJFTE	SCALE087
	YI=C6*EDGEI	SCALE088
C*		SCALE089
C*	SCAN UU TABLE FOR BOUNDARIES OF MIXING ZONES	SCALE090
C*		SCALE091
	DO 220 L=1,NPU	SCALE092
	IF(UU(L).NF, UCL1) GO TO 222	SCALE093
	220 CONTINUE	SCALE094
	222 LK=L=1	SCALE095
	REFL=YI-C6*Y(LK)	SCALE096
	DO 225 L=LK,NPU	SCALE097
	IF(UU(L).GT,TEST1 .AND. UU(L).LT, TEST2) UU(L)=110	SCALE098
	IF(UU(L).EQ,U0) LJ=L	SCALE099
		SCALE100

225 CONTINUE	SCALE101
C* 230 REFL0=Y0-C6*Y(LJ)	SCALE102
GO TO 500	SCALE103
C* 230 REFL0=Y0-C6*Y(LJ)	SCALE104
C* DOWN STREAM-- DETERMINE BOUNDARIES OF MIXING ZONES//	SCALE105
C* USE LINEAR EQUATIONS FOR NOZZLE/MERGE POINT LINES	SCALE106
260 YI=(SLOPED*X+CEPTD) *C6	SCALE107
Y0=C6*YJETF	SCALE108
REFLO=Y0-C6*(SLOPEI*X+CEPTI)	SCALE109
DO 270 L=1,NPU	SCALE110
IF(UU(L).NE. 1.) GO TO 275	SCALE111
270 CONTINUE	SCALE112
C* 275 REFL=YI-C6*Y(L-1)	SCALE113
C* 500 RETURN	SCALE114
END	SCALE115
	SCALE116
	SCALE117
	SCALE118

C	SCKP	MAIN DRIVER FOR SCKP CALCULATION	SCKP0001
		SUBROUTINE SCKP(KK)	SCKP0002
		COMMON /GASCOMP/ RICH(12,2),FUEL(2,12,2),FNTH(2,12,2),	SCKP0003
		CONC(16,2,12,2),HCINCP(12,2),OTHER(192)	SCKP0004
		COMMON /GASTMW/ TG(2,12,2),MWTG(2,12,2),TAU(12,2),CPG(2,12,2)	SCKP0005
		REAL MWTG	SCKP0006
		COMMON /PSFQ / FQA,8FQS,TP,X(16),DHDDMW,TEQ,8FQ,XMWT,MNEQ	SCKP0007
		COMMON /CBITS / BITS,BLANK	SCKP0008
		COMMON /CKINET/ DUMC1(4),XMOLW0,DUMSK(90),ENTRY1	SCKP0009
		LOGICAL ENTRY1	SCKP0010
		COMMON /TNDLF/ XIN(12)	SCKP0011
		COMMON /TNDATA/ HQC1,HF,WAR,T2,BTA,T25,FARS,FINO2C,P0,DUM2(88)	SCKP0012
		COMMON /PSFQX / HQC,HUM,COPATR,MAIR,FS,FUELMW	SCKP0013
		REAL MAIR	SCKP0014
		COMMON /SNMW / ALSP(150),WMT(75)	SCKP0015
		COMMON /GHSC / FF(25),HZ(25),DUMG(75)	SCKP0016
		COMMON /CPRINT/ PDUM(20)	SCKP0017
		DIMENSION C1(16),C2(16)	SCKP0018
		DATA R0/1,98596/	SCKP0019
C			SCKP0020
		HQC = HQC1	SCKP0021
		HUM = WAR	SCKP0022
		FUELMW = 12.01 + 1.008 * HQC	SCKP0023
		FS = FUELMW / (1. + .25 * HQC) * .209495 / MAIR	SCKP0024
		K = KK	SCKP0025
		WRITE (6,100) K	SCKP0026
100		FORMAT(//8X,10HSTREAMTUBE,2X,12//)	SCKP0027
		TP = TG(1,K,2)	SCKP0028
		IF(TP,LE,1500.) GO TO 4	SCKP0029
		TPK = TP / 1.8	SCKP0030
		CALL THRM(TPK,1.)	SCKP0031
		CALL SETM(2,0.,C1,16,C2,16)	SCKP0032
		Y2 = 0.	SCKP0033
		DO 90 I=12,16	SCKP0034
90		Y2 = Y2 + WMT(I) * CONC(I,1,K,2)	SCKP0035
		QY2 = 0.	SCKP0036
		IF(Y2.EQ.0.) GO TO 92	SCKP0037
		QY2 = 1./Y2	SCKP0038
92		QYY2 = 1./(1.-Y2)	SCKP0039
C			SCKP0040
C		SPLIT MIXTURE TO CONSERVE ENERGY	SCKP0041
C			SCKP0042
		CALL FMPYC(1,QY2,CONC(12,1,K,2),C2(12),5)	SCKP0043
		CALL FMPYC(1,QYY2,CONC(1,1,K,2),C1,1)	SCKP0044
		HS1 = 0.	SCKP0045
		HS2 = 0.	SCKP0046
		TMOLFS = 0.	SCKP0047
		DO 91 I=1,16	SCKP0048
		TMOLFS = TMOLFS + C1(I)	SCKP0049
		HS1 = HS1 + C1(I) * HZ(I)	SCKP0050

	MS2 = MS2+C2(I)*HZ(I)	SCKP0051
91	CONTINUE	SCKP0052
	MS1 = R0*TP*MS1	SCKP0053
	MS2 = R0*TP*MS2	SCKP0054
	QTMOL = 1./TMOLFS	SCKP0055
	CALL FMPYC(1,QTMOL,C1,XIN,11)	SCKP0056
	M = MS1	SCKP0057
	IF(PDUM(10).NE.0.) CALL TABPRT(5HXINIT,XIN,11,10)	SCKP0058
C		SCKP0059
C	CALCULATE EQUIVALENCE RATIO	SCKP0060
C		SCKP0061
	ZETA1 = 2.*XIN(9)+XIN(11)	SCKP0062
	ZETA = XIN(8)*.5*CO2AIR/.780881+ZETA1	SCKP0063
	FR = 3.*.780881/.209495*(XIN(7)+ZETA)/ZETA1	SCKP0064
	FOA1 = FS*ER	SCKP0065
	DTIME = TAU(K,2)-TAU(K,1)	SCKP0066
C		SCKP0067
C	CALL FOR KINETICS STEP	SCKP0068
C		SCKP0069
	ENTRY1 = .TRUE.	SCKP0070
	XMOLW0 = RITS	SCKP0071
	2 CALL KINET(P0,H,FOA1,0.)	SCKP0072
C		SCKP0073
	CALL KINET(P0,H,FOA1,DTIME)	SCKP0074
C		SCKP0075
C	STORE THE NEW CONCENTRATIONS	SCKP0076
C		SCKP0077
	IF(PDUM(10).NE.0.) CALL TABPRT(6HT,XFIN,TP,12,10)	SCKP0078
3	TERM = 1./((XMWT*(1.-Y2))	SCKP0079
	CALL FMPYC(1,TERM,X,CONC(1,1,K,1),11)	SCKP0080
	CALL FMPYC(1,Y2,C2(12),CONC(12,1,K,1),5)	SCKP0081
	ENTH(1,K,1) = MS1*(1.-Y2)+MS2*Y2	SCKP0082
	TG(1,K,1) = TP	SCKP0083
	GO TO 5	SCKP0084
4	TG(1,K,1) = TG(1,K,2)	SCKP0085
	ENTH(1,K,1) = ENTH(1,K,2)	SCKP0086
	CALL MOVE(1,CONC(1,1,K,2),CONC(1,1,K,1),16,1)	SCKP0087
C		SCKP0088
5	RETURN	SCKP0089
	END	SCKP0090

CSERCH OPTIMAL SEQUENTIAL SEARCH TECHNIQUE
SUBROUTINE SEARCH (F,FOFX,A,X,B)

C		SERCH001
C		SERCH002
C		SERCH003
C	THIS ROUTINE USES AN OPTIMAL SEQUENTIAL SEARCH TECHNIQUE TO FIND X	SERCH004
C	IN (A,B) SUCH THAT F(X) = FOFX	SERCH005
C		SERCH006
C	ASSUMPTIONS	SERCH007
C	(1) F(X) CONTINUOUS ON THE CLOSED INTERVAL (A,B)	SERCH008
C	(2) FOFX IS NOT EQUAL TO ZERO	SERCH009
C		SERCH010
C	COMMON/SINT/DUM(10),KOUNT,FX	SERCH011
C		SERCH012
	X1 = (2.*A + B)/3.	SERCH013
	KOUNT = 1	SERCH014
	F1 = F(X1)	SERCH015
	IF (ABS(1.-F1/FOFX) .GT. 0.0001) GO TO 2	SERCH016
	X = X1	SERCH017
	FX = F1	SERCH018
	RETURN	SERCH019
C		SERCH020
	2 X2 = (A + 2.*B)/3.	SERCH021
	KOUNT = KOUNT + 1	SERCH022
	F2 = F(X2)	SERCH023
	IF (ABS(1.-F2/FOFX) .GT. 0.0001) GO TO 3	SERCH024
	X = X2	SERCH025
	FX = F2	SERCH026
	RETURN	SERCH027
C		SERCH028
	3 D1 = F1 - FOFX	SERCH029
	IF (D1*(FOFX-F2) .GT. 0.) GO TO 5	SERCH030
	IF (D1 .LT. 0.) GO TO 4	SERCH031
	X2 = A	SERCH032
	KOUNT = KOUNT + 1	SERCH033
	F2 = F(X2)	SERCH034
	IF (F2 .LT. FOFX .AND. ABS(1.-F2/FOFX) .GT. 0.0001) GO TO 5	SERCH035
	X = A	SERCH036
	FX = F2	SERCH037
	RETURN	SERCH038
C		SERCH039
	4 X1 = B	SERCH040
	KOUNT = KOUNT + 1	SERCH041
	F1 = F(X1)	SERCH042
	IF (F1 .GT. FOFX .AND. ABS(1.-F1/FOFX) .GT. 0.0001) GO TO 5	SERCH043
	X = B	SERCH044
	FX = F1	SERCH045
	RETURN	SERCH046
C		SERCH047
	5 X = (X1 + X2)/2.	SERCH048
	KOUNT = KOUNT + 1	SERCH049
	FX = F(X)	SERCH050

```

IF (ABS(1.-FX/FOFX) ,LE. 0.0001) RETURN
IF (KOUNT .EQ. 11) RETURN
IF ((FX-FOFX)*(FOFX-F2) .LT. 0.) GO TO 6
X1 = X
GO TO 5
6 X2 = X
F2 = FX
GO TO 5
c
END

```

```

SERCH051
SERCH052
SERCH053
SERCH054
SERCH055
SERCH056
SERCH057
SERCH058
SERCH059
SERCH060

```

CSETUP	SETS-UP FLOW PROPERTIES FOR INTERPOLATION	SETUP001
C		SETUP002
	SUBROUTINE SETUP (IRST,K)	SETUP003
C		SETUP004
	COMMON /CPROLI/ G(50,1),ALPHA(1,1),BETA(50,1),GAMM(50,1),	SETUP005
	1 DELTA(50,1)	SETUP006
	COMMON /CINIT/ XX,DPSI(1)	SETUP007
	COMMON /CSTART/ GJET(200),GEX	SETUP008
	COMMON /CCOUNT/ X(5),NSL(5),NX,XMAX	SETUP009
	COMMON /CINPUT/ PSI(100,5),R(100,5),U(100,5),E(100,5),RHO(100,5),	SETUP010
	1 XLN(100,5),F(100,5)	SETUP011
	COMMON /CNUNET/ PSIMAX(5),NSLZ(5),RZ(50,5),UZ(50,5),EZ(50,5),	SETUP012
	1 RHOZ(50,5),XLNZ(50,5),FZ(50,5)	SETUP013
	COMMON /CTRANS/ RZT(5,50),UZT(5,50),EZT(5,50),RHOZT(5,50),	SETUP014
	1 XLNZT(5,50),FZT(5,50)	SETUP015
	COMMON /CSIDAT/ SY(50),RAD(50),VEL(50),TKE(50),DFN(50),ILS(50),	SETUP016
	1 FAR(50)	SETUP017
	COMMON /CADENT/ MAX,NSI,NXS,NSZMAX,NSZI	SETUP018
C*		SETUP019
C*	DETERMINE VALUES OF STREAM FUNCTION ON STREAMLINES AT INPUT STATIONS	SETUP020
	DO 170 I=1,NX	SETUP021
	NS=NSL(I)	SETUP022
	PSIMAX(I)=PSI(NS,I)	SETUP023
	SLZ=PSIMAX(I)/DPSI(1)	SETUP024
	NSLZ(I)=IFIX(SLZ)+1	SETUP025
	IF (NSLZ(I),GT,MAX) GO TO 180	SETUP026
170	CONTINUE	SETUP027
	NXS=NX	SETUP028
	GO TO 190	SETUP029
180	NXS=1	SETUP030
	NSLZ(NXS)=MAX	SETUP031
190	SY(1)=0.	SETUP032
	NSZMAX=NSLZ(NXS)	SETUP033
	DO 200 J=2,NSZMAX	SETUP034
	SY(J)=SY(J-1)+DPSI(1)	SETUP035
200	CONTINUE	SETUP036
C*	CALCULATE FLUID PROPERTIES ON STREAMLINES AT INPUT STATIONS	SETUP037
	DO 205 I=1,NXS	SETUP038
	NS=NSL(I)	SETUP039
	NSZ=NSLZ(I)	SETUP040
	CALL LFITI (PSI(1,I),R(1,I),NS,SY,RZ(1,I),NSZ,0)	SETUP041
	CALL LFITI (PSI(1,I),U(1,I),NS,SY,UZ(1,I),NSZ,0)	SETUP042
	CALL LFITI (PSI(1,I),E(1,I),NS,SY,EZ(1,I),NSZ,0)	SETUP043
	CALL LFITI (PSI(1,I),RHO(1,I),NS,SY,RHOZ(1,I),NSZ,0)	SETUP044
	CALL LFITI (PSI(1,I),XLN(1,I),NS,SY,XLNZ(1,I),NSZ,0)	SETUP045
	CALL LFITI (PSI(1,I),F(1,I),NS,SY,FZ(1,I),NSZ,0)	SETUP046
205	CONTINUE	SETUP047
	DO 215 IR=1,NXS	SETUP048
	I=NXS+1-IR	SETUP049
	NSZ=NSLZ(I)	SETUP050

NSZP1=NSLZ(I)+1	SETUP051
IF (NSZP1.GT.NSZMAX) GO TO 215	SETUP052
DO 210 J=NSZP1,NSZMAX	SETUP053
RZ(J,I)=RZ(J,I+1)*RZ(NSZ,I)/RZ(NSZ,I+1)	SETUP054
UZ(J,I)=U(NS,I)	SETUP055
EZ(J,I)=E(NS,I)	SETUP056
RHOZ(J,I)=RHO(NS,I)	SETUP057
XLNZ(J,I)=XLN(NS,I)	SETUP058
FZ(J,I)=F(NS,I)	SETUP059
210 CONTINUE	SETUP060
215 CONTINUE	SETUP061
IF (IRST.GT.0) GO TO 220	SETUP062
NSZ1=NSLZ(1)	SETUP063
IF (UZ(NSZ1,1).LT.(0.0)) UZ(NSZ1,1)=0.	SETUP064
IF (EZ(NSZ1,1).LT.(0.0)) EZ(NSZ1,1)=0.	SETUP065
IF (FZ(NSZ1,1).LT.(0.0)) FZ(NSZ1,1)=0.	SETUP066
CALL LFIT1 (PSI,GJET,NS1,SY,G,NSZ1,0)	SETUP067
IF (G(NSZ1,K).LT.(0.0)) G(NSZ1,K)=0.	SETUP068
220 CONTINUE	SETUP069
C* TRANSPOSE MATRICES FOR INTERPOLATION ALONG STREAMLINES	SETUP070
DO 240 I=1,NXS	SETUP071
DO 230 J=1,NSZMAX	SETUP072
RZT(I,J)=RZ(J,I)	SETUP073
UZT(I,J)=UZ(J,I)	SETUP074
EZT(I,J)=EZ(J,I)	SETUP075
RHOZT(I,J)=RHOZ(J,I)	SETUP076
XLNZT(I,J)=XLNZ(J,I)	SETUP077
FZT(I,J)=FZ(J,I)	SETUP078
230 CONTINUE	SETUP079
240 CONTINUE	SETUP080
RETURN	SETUP081
END	SETUP082

CSOLV3	SOLVE 3 LINEAR EQUATIONS $A(I,J)*X(J)=B(I)$	SOLV3001
	SUBROUTINE SOLV3(A,B,X,NOSOLN)	SOLV3002
	LOGICAL NOSOLN	SOLV3003
	DIMENSION A(3,3),B(3),X(3),D(3,3)	SOLV3004
	C=0.0	SOLV3005
	DO 20 I=1,3	SOLV3006
	DO 10 J=1,3	SOLV3007
	I1=I+1	SOLV3008
	IF(I1.GT.3) I1=I+3	SOLV3009
	I2=I+2	SOLV3010
	IF(I2.GT.3) I2=I+3	SOLV3011
	J1=J+1	SOLV3012
	IF(J1.GT.3) J1=J+3	SOLV3013
	J2=J+2	SOLV3014
	IF(J2.GT.3) J2=J+3	SOLV3015
10	D(I,J)=A(I1,J1)*A(I2,J2)+A(I1,J2)*A(I2,J1)	SOLV3016
20	C=C+D(I,3)*A(I,3)	SOLV3017
	NOSOLN=C.EQ.0.0	SOLV3018
	IF(NOSOLN)RETURN	SOLV3019
	DO 40 J=1,3	SOLV3020
	X(J)=0.0	SOLV3021
	DO 30 I=1,3	SOLV3022
30	X(J)=X(J)+D(I,J)*B(I)	SOLV3023
40	X(J)=X(J)/C	SOLV3024
	RETURN	SOLV3025
	END	SOLV3026

CSPALD	-SPALD- MAIN SUBROUTINE FOR CALC. OF -G-	SPALD001
C		SPALD002
C*	OCTOBER 1975 MODIFICATIONS FOR AUG FMIS SYS	SPALD003
C*		SPALD004
	SUBROUTINE SPALD	SPALD005
C		SPALD006
C*	DIAJ = JET DIAMETER -- IN	SPALD007
C*	VJET = JET VELOCITY -- FT/SEC	SPALD008
C*	EJET = JET TKE -- RTU/LRM	SPALD009
C*	GJET = INITIAL DISTRIBUTION OF -G- IN JET	SPALD010
C*	GEX = VALUE OF -G- IN EXTERNAL FLOW	SPALD011
C*	GCJ = DIMENSIONAL CONSTANT -- LBM-FT**2/RTU-SEC**2	SPALD012
C*	PSI = STREAM FUNCTION -- LBM/FT**3	SPALD013
C*	R = DIMENSIONLESS RADIUS	SPALD014
C*	U = DIMENSIONLESS VELOCITY	SPALD015
C*	E = DIMENSIONLESS TKE	SPALD016
C*	F = MEAN FUEL/AIR RATIO	SPALD017
C*	RHO = DENSITY -- LBM/FT**3	SPALD018
C*	XLN = TURBULENCE LENGTH SCALE -- FT	SPALD019
C		SPALD020
	LOGICAL PRINT	SPALD021
	COMMON /CTHETA/ THETA,II	SPALD022
	COMMON /CENDS/ JSTART,JEND	SPALD023
	COMMON /CPBULI/ G(50,1),ALPHA(1,1),BETA(50,1),GAMM(50,1),	SPALD024
	1 DELTA(50,1)	SPALD025
	COMMON /CTRIDI/ COEFL(50),COEFC(50),COEFR(50),RHS(50)	SPALD026
	COMMON /CBNDRY/ DX,DXPRN,JT	SPALD027
	COMMON /CINIT/ XX,DPSI(1)	SPALD028
	COMMON /CJETOT/ GCJ,DIAJ,VJET,EJET,NXTA	SPALD029
	COMMON /CSTART/ GJET(200),GEX	SPALD030
	COMMON /CCOUNT/ X(5),NSL(5),NX,XMAX	SPALD031
	COMMON /CINPUT/ PSI(100,5),R(100,5),U(100,5),E(100,5),RHO(100,5),	SPALD032
	1 XLN(100,5),F(100,5)	SPALD033
	COMMON /CCONST/ CONST1,CONST2,CONST3,CONST4	SPALD034
	COMMON /CNUDET/ PSIMAX(5),NSLZ(5),RZ(50,5),UZ(50,5),EZ(50,5),	SPALD035
	1 RHOZ(50,5),XLNZ(50,5),FZ(50,5)	SPALD036
	COMMON /CTRANS/ RZT(5,50),UZT(5,50),FZT(5,50),RHOZT(5,50),	SPALD037
	1 XLNZT(5,50),FZT(5,50)	SPALD038
	COMMON /CSLDAT/ SY(50),RAD(50),VEL(50),TKE(50),DEN(50),TLS(50),	SPALD039
	1 FAR(50)	SPALD040
	COMMON /CADCNT/ MAX,NS1,NXS,NS2MAX,NSZ1	SPALD041
	COMMON /CPREF/ DUMH(2414),XW(100)	SPALD042
	COMMON /CFILK / CSC	SPALD043
C		SPALD044
	COMMON / KEYS / KEYS(11), KEYS(11), KODA(11), KODB(11)	SPALD045
C		SPALD046
	DIMENSION LGSPG(5), NSSPG(6), NS2SPG(2)	SPALD047
C		SPALD048
	DATA LGSPG/2, 50, 1, 50, 0/	SPALD049
	DATA NSSPG/1, 4HPSI, 1, 2HIR, 1, 2HIG/	SPALD050

C	DATA NS2SPG/1, 3HIGK/	SPALD051
		SPALD052
	NAMLIST /CHANGE/ ALFA,CG1,CG2,SIGMAG,THETA,DX,DXPRN,XSTOP,NSZ1	SPALD053
	* ,PRINT	SPALD054
C		SPALD055
C		SPALD056
	PRINT = .FALSE.	SPALD057
	XSTOP = 1000.	SPALD058
	DX = 1000.	SPALD059
	READ (5,CHANGE)	SPALD060
C		SPALD061
	IENTRY=0	SPALD062
	CALL READI (IENTRY)	SPALD063
	IENTRY=IENTRY+NX-1	SPALD064
	JSTART=1	SPALD065
	II=1	SPALD066
	JT=1	SPALD067
	K=1	SPALD068
	IX=1	SPALD069
	IRST=0	SPALD070
	MAX=50	SPALD071
	ALPHA(1,1)=1./((VJET+DIAJ/12.))	SPALD072
	CONST1=ALFA*SQRT(GCJA*EJET)	SPALD073
	CONST2=CG1/(VJET+DIAJ/12.)	SPALD074
	CONST3=CG2*(DIAJ/12.)/(ALFA*ALFA*VJET)	SPALD075
	CONST4=1./SIGMAG	SPALD076
	IF (XSTOP.GT.XMAX) XSTOP=XMAX	SPALD077
C*	SET INITIAL STREAMLINE SPACING	SPALD078
	NS1=NSL(1)	SPALD079
	DO 150 J=1,NS1	SPALD080
	IF (R(J,1).LT.1.0) GO TO 150	SPALD081
	DPSI(1)=PSI(J,1)/(FLOAT(NSZ1-1))	SPALD082
	GO TO 160	SPALD083
	150 CONTINUE	SPALD084
	DPSI(1)=PSI(NS1,1)/(FLOAT(NSZ1-1))	SPALD085
C*	SET-UP FLOW PROPERTIES FOR INTERPOLATION ALONG STREAMLINES	SPALD086
	160 CALL SETUP (IRST,K)	SPALD087
C*	PRINT -G- PROFILE AT INITIAL STATION	SPALD088
	XX=0.	SPALD089
	JFND=NSZ1	SPALD090
	CALL STPRUP (K)	SPALD091
	KEYB(7)=CSC*XX+0.5	SPALD092
	WRITE (3) KXX,SY,RAD,G,JEND	SPALD093
	IF(.NOT.PRINT) GO TO 170	SPALD094
	CALL TABPRT (6H* X=,XX,IX,1)	SPALD095
	CALL TABPRT (5H PSI,SY,JEND,10)	SPALD096
	CALL TABPRT (3H R,RAD,JEND,10)	SPALD097
	CALL TABPRT (3H G,G,JEND,10)	SPALD098
C*	MOVE DOWNSTREAM FROM INITIAL STATION	SPALD099
	170 SYMAX = 0.	SPALD100

SYNEXI=SY(NSZ1)+DPSI(1)	SPALD101
C* DO-LOOP FOR DOWNSTREAM CALCULATIONS	SPALD102
XD = XW(2)	SPALD103
DX = AMIN1(DX,XD-XW(1))	SPALD104
IP = 2	SPALD105
180 XX = XX+DX	SPALD106
ISPACE=0	SPALD107
C*	SPALD108
C*	SPALD109
C* CHECK TO SEE IF ADDITIONAL INTERPOLATION STATIONS ARE REQUIRED	SPALD110
IF (XX.LE.X(NX)) GO TO 270	SPALD111
C* IF REQUIRED, ADD DOWNSTREAM INTERPOLATION STATIONS	SPALD112
CALL MOVE(5,X(5),X(1),1,1,NSL(5),NSL(1),1,1,PSI(1,5),PSI(1,1),	SPALD113
* 100,1,R(1,5),R(1,1),100,1,U(1,5),U(1,1),100,1)	SPALD114
CALL MOVE(4,E(1,5),E(1,1),100,1,RHO(1,5),RHO(1,1),100,1,	SPALD115
* XLN(1,5),XLN(1,1),100,1,F(1,5),F(1,1),100,1)	SPALD116
CALL SETM(5,0,X(2),4,NSL(2),4,PSI(1,2),400,R(1,2),400,U(1,2),400)	SPALD117
CALL SETM(5,0,E(1,2),400,RHO(1,2),400,XLN(1,2),400,F(1,2),400,	SPALD118
* PSIMAX,5)	SPALD119
CALL SETM(5,0,RZ,250,UZ,250,EZ,250,RHOZ,250,XLNZ,250)	SPALD120
CALL SETM(5,0,FZ,250,RZT,250,UZT,250,EZT,250,RHOZT,250)	SPALD121
CALL SETM(2,0,XLNZT,250,FZT,250)	SPALD122
CALL SETM(1,0,NSLZ,5)	SPALD123
CALL READT (IENTRY)	SPALD124
IENTRY=IENTRY+NX-1	SPALD125
C* SET-UP FLOW PROPERTIES FOR INTERPOLATION THROUGH NEW STATIONS	SPALD126
IRST=IRST+1	SPALD127
CALL SETUP (IRST,K)	SPALD128
270 CONTINUE	SPALD129
C*	SPALD130
C*	SPALD131
C* CHECK TO SEE IF ANOTHER STREAMLINE SHOULD BE ADDED	SPALD132
CALL LFIT1 (X,PSIMAX,NX,XX,SYMAX,IX,0)	SPALD133
IF (SYMAX.LT.SYNEXT) GO TO 320	SPALD134
C* IF REQUIRED, ADD ANOTHER STREAMLINE	SPALD135
JEND=JEND+1	SPALD136
C* CHECK TO SEE IF NUMBER OF STREAMLINES EXCEEDS LIMIT	SPALD137
IF (JEND.LE.MAX) GO TO 300	SPALD138
C* IF REQUIRED, RE-SET STREAMLINE SPACING	SPALD139
ISPACE=1	SPALD140
DPSI(1)=DPSI(1)+DPSI(1)	SPALD141
CALL SETM(5,0,RZ,250,UZ,250,EZ,250,RHOZ,250,XLNZ,250)	SPALD142
CALL SETM(5,0,FZ,250,RZT,250,UZT,250,FZT,250,RHOZT,250)	SPALD143
CALL SETM(5,0,XLNZT,250,FZT,250,SY,50,RAD,50,VFL,50)	SPALD144
CALL SETM(5,0,TKE,50,DEN,50,TL5,50,FAR,50,BETA,50)	SPALD145
CALL SETM(2,0,GAMM,50,DELTA,50)	SPALD146
C* SET-UP FLOW PROPERTIES FOR INTERPOLATION ALONG REVISED STREAMLINES	SPALD147
IRST=IRST+1	SPALD148
CALL SETUP (IRST,K)	SPALD149
NEWJN=(JEND+1)/2	SPALD150

DO 290 J=1,NEWJN	SPALD151
JJ=(2*J)-1	SPALD152
IF (JJ.GE,JEND) GO TO 280	SPALD153
G(J,K)=G(JJ,K)	SPALD154
GO TO 290	SPALD155
280 G(J,K)=GEX	SPALD156
290 CONTINUE	SPALD157
NSTART=NEWJN+1	SPALD158
NG=MAX-NEWJN	SPALD159
CALL SETM (1,0,0,G(NSTART,K),NG)	SPALD160
JEND=NEWJN	SPALD161
GO TO 310	SPALD162
300 G(JEND,K)=GEX	SPALD163
JSTART=JEND	SPALD164
C* RE-CALC. PDE COEFFICIENTS AT PREVIOUS STATION ON REVISED STREAMLINES	SPALD165
310 XX=XX-DX	SPALD166
CALL STPROP (K)	SPALD167
CALL COEFF(K)	SPALD168
JSTART=1	SPALD169
XX=XX+DX	SPALD170
SYNEXT=SY(JEND)+DPS1(1)	SPALD171
320 CONTINUE	SPALD172
C*	SPALD173
C*	SPALD174
C* CALCULATE FLUID PROPERTIES ON ALL STREAMLINES AT NEW STATION	SPALD175
CALL STPROP (K)	SPALD176
C* SOLVE PARABOLIC PDE AT EACH GRID POINT AND PRINT RESULTS	SPALD177
CALL PROLIC(K)	SPALD178
IF (XX.LT,XD) GO TO 340	SPALD179
C* WRITE DATA RECORDS - - AT EACH XX - -	SPALD180
C* SY - - RAD - - G	SPALD181
C*	SPALD182
LGSPG(2) = JEND	SPALD183
LGSPG(4) = JEND	SPALD184
KEY8(7) = CSC * XX + 0.5	SPALD185
KXX = KEY8(7)	SPALD186
WRITE (3) KXX,SY,RAD,G,JEND	SPALD187
C	SPALD188
IF (.NOT.PRINT) GO TO 330	SPALD189
WRITE (6,321) XX	SPALD190
321 FORMAT(7X,2HX=F16.6/)	SPALD191
WRITE (6,322)	SPALD192
322 FORMAT(//5X,3HPSI,9X,1HR,11X,1HG,11X,3HPSI,9X,1HR,11X,1HG/)	SPALD193
DO 323 I=1,JEND,2	SPALD194
WRITE (6,324) SY(I),RAD(I),G(I,1),SY(I+1),RAD(I+1),G(I+1,1)	SPALD195
323 CONTINUE	SPALD196
324 FORMAT(2(F10.6,F12.6,E13.4))	SPALD197
330 IP = IP+1	SPALD198
IF (IP.GT,NXTA) GO TO 400	SPALD199
XD = X*(IP)	SPALD200

```

      DX      = DXSAVE
C* RE=SET AXIAL SPACING WHEN STREAMLINE SPACING HAS BEEN RE=SET
340 IF( ISPACE.EQ.1 ) DX=DX+DX
      DXSAVE = DX
      DX      = AMINI( DX, XD-XX )
      GO TO 180
400 RETURN
      END

```

```

SPALD201
SPALD202
SPALD203
SPALD204
SPALD205
SPALD206
SPALD207
SPALD208

```

CSTPROP CALCULATES FLUID PROPERTIES AT EACH POINT

SUBROUTINE STPROP (K)

COMMON /CENDS/ JSTART,JEND

COMMON /CINIT/ XX,DPSI(1)

COMMON /CCOUNT/ X(5),NSL(5),NX,XMAX

COMMON /CTRANS/ RZT(5,50),UZT(5,50),FZT(5,50),RHOZT(5,50),

1 XLNZT(5,50),FZT(5,50)

COMMON /CSTAT/ SY(50),RAD(50),VFL(50),TKE(50),DEN(50),TIS(50),

1 FAR(50)

COMMON /CADCNT/ MAX,NSI,NXS,NSZMAX,NSZI

IX=1

DO 330 J=JSTART,JEND

CALL LFIT1 (X,RZT(1,J),NXS,XX,RAD(J),IX,0)

CALL LFIT1 (X,UZT(1,J),NXS,XX,VEL(J),IX,0)

CALL LFIT1 (X,FZT(1,J),NXS,XX,TKE(J),IX,0)

CALL LFIT1 (X,RHOZT(1,J),NXS,XX,DEN(J),IX,0)

CALL LFIT1 (X,XLNZT(1,J),NXS,XX,TLS(J),IX,0)

CALL LFIT1 (X,FZT(1,J),NXS,XX,FAR(J),IX,0)

IF (VEL(J).LT.(0.0)) VEL(J)=0.0

IF (TKE(J).LT.(0.0)) TKE(J)=0.0

IF (FAR(J).LT.(0.0)) FAR(J)=0.0

330 CONTINUE

RETURN

END

STPROP01

STPROP02

STPROP03

STPROP04

STPROP05

STPROP06

STPROP07

STPROP08

STPROP09

STPROP10

STPROP11

STPROP12

STPROP13

STPROP14

STPROP15

STPROP16

STPROP17

STPROP18

STPROP19

STPROP20

STPROP21

STPROP22

STPROP23

STPROP24

STPROP25

STPROP26

STPROP27

CSUMCPD	DIFFUSION FLUX TERM FOR ENERGY EQUATION	SUMCPD01
FUNCTION	SUMCPD(L)	SUMCPD02
C*****		SUMCPD03
C*****	SPECIAL VERSION FOR 12 SPECIES	SUMCPD04
C*****		SUMCPD05
	COMMON /ERASE/ DUMS1(400), T(200), DIMS2(200)	SUMCPD06
	COMMON /DIFER/	SUMCPD07
	* NC , CNAME(12) , ALJ(12) , ALJO(12) , ALE(12) , SCH(12) ,	SUMCPD08
	* TCRPF(12) , HCRPF(12) , CPC(3,12)	SUMCPD09
	COMMON /MDI UP/ ALXU(1200),DALXU(100,12),DUME(200)	SUMCPD10
	DIMENSION CPT(12)	SUMCPD11
C*		SUMCPD12
C*		SUMCPD13
C*		SUMCPD14
	1 TEMP=T(L)	SUMCPD15
	DD 10 K=1,NC	SUMCPD16
	10 CPT(K)=CPC(1,K)+CPC(2,K)*TEMP+CPC(3,K)*TEMP*TEMP	SUMCPD17
C*		SUMCPD18
C*		SUMCPD19
C*	MULTIPLY BY GRADIENTS/SCHMIDT NUMBERS AND SUM	SUMCPD20
C*		SUMCPD21
	12 SUMCPD=0.	SUMCPD22
	DD 15 K=1,NC	SUMCPD23
	15 SUMCPD=SUMCPD+CPT(K)/SCH(K)*DALXU(L,K)	SUMCPD24
	20 RETURN	SUMCPD25
	END	SUMCPD26

C	SUMP	CALC. AND PRINT RESIDUAL EMISSIONS INDICES	SUMP0001
C		AT SFELECTED AXIAL POSNS.	SUMP0002
C			SUMP0003
C		SUBROUTINE SUMUP	SUMP0004
C			SUMP0005
		REAL MWT(2),MWTBAR,M(2)	SUMP0006
		INTEGER DAY,YEAR	SUMP0007
		DIMENSION SPV(2),CP(2),X(16),T(2),OPT(10,11,3),SUM(4),GAM(2),	SUMP0008
		IFGAM(2)	SUMP0009
C			SUMP0010
		COMMON/GASCOMP/YY(12,2),F(2,12,2),H(2,12,2),Z(16,2,12,2),	SUMP0011
		* DUM1(24),DUM2(192)	SUMP0012
		COMMON/DPCTRL/TITLE(20),PRINT(30)/JETDAT/NT,RAD(12),DUM3(132)	SUMP0013
		COMMON/CAAXIAL/DIST,DUM4(2)	SUMP0014
		COMMON/CMASS / DUM5(12),K(12),DUM6(24)	SUMP0015
		COMMON/INDATA/HC,HF,HAK,DUM7(5),PO,DUM8(88)	SUMP0016
		COMMON /STCTRL / LSTA,DUMST(16)	SUMP0017
		COMMON /CBITS/ BITS,BLANK	SUMP0018
C			SUMP0019
C			SUMP0020
		AVG(X1,X2)=YYY*X1+(1.0-YYY)*X2	SUMP0021
C			SUMP0022
		CALL SFTM(1,BITS,OPT,330)	SUMP0023
		CALL SFTM(1,0.0,SUM,4)	SUMP0024
		DO 70 N=1,NT	SUMP0025
C			SUMP0026
		COMPUTE PROPERTIES OF RICH (1) AND LEAN (2) PARTS	SUMP0027
C			SUMP0028
		DO 20 IP=1,2	SUMP0029
		MWT(IP)=0.0	SUMP0030
		DO 10 J=1,16	SUMP0031
10		MWT(IP)=MWT(IP)+Z(I,IP,N,1)	SUMP0032
		MWT(IP)=1.0/MWT(IP)	SUMP0033
		CALL MOVE(1,Z(I,IP,N,1),X,16,1)	SUMP0034
		CALL TFM41(X,H(IP,N,1),I(IP),CP(IP))	SUMP0035
		SPV(IP)=10.732459/MWT(IP)*T(IP)/PO	SUMP0036
		GAM(IP)=CP(IP)*MWT(IP)/1.98596	SUMP0037
20		GAM(IP)=GAM(IP)/(GAM(IP)-1.0)	SUMP0038
C			SUMP0039
C		FREE STREAM MEAN PROPERTIES	SUMP0040
C			SUMP0041
		ASSIGN 50 TO MEAN	SUMP0042
		KK=2	SUMP0043
		YYY=YY(N,1)	SUMP0044
C			SUMP0045
30		FBAR=AVG(F(1,N,1),F(2,N,1))	SUMP0046
		SPVBAR=AVG(SPV(1),SPV(2))	SUMP0047
		MWTBAR=1.0/AVG(1.0/MWT(1),1.0/MWT(2))	SUMP0048
		DO 40 J=1,16	SUMP0049
40		X(I)=AVG(Z(I,1,N,1),Z(I,2,N,1))	SUMP0050

FICD=28.01*X(7)/FBAR	SUMP0051
FIMC=(120.1+10.08*HC)*X(16)/FBAR	SUMP0052
FINDX=45.009*(X(11)+X(14))/FBAR	SUMP0053
CALL FMPYC(1,MWTRAR,X,X,16)	SUMP0054
DRYMOL=1.006202*(1.0-X(6))	SUMP0055
IF(DRYMOL.GT.1.0)DRYMOL=1.0	SUMP0056
RCN=X(7)/DRYMOL	SUMP0057
RHC=10.0*X(16)	SUMP0058
RNOX=X(11)+X(14)	SUMP0059
C	SUMP0060
C PLACE PROPERTIES IN OUTPUT VECTOR	SUMP0061
C	SUMP0062
OPT(1,N,KK)=YYY	SUMP0063
OPT(2,N,KK)=FBAR*(1.0+WAR)/(1.0-FBAR)	SUMP0064
OPT(3,N,KK)=1E3*EICD	SUMP0065
OPT(4,N,KK)=1E3*FIMC	SUMP0066
OPT(5,N,KK)=1E3*FINDX	SUMP0067
OPT(6,N,KK)=1E6*RCN	SUMP0068
OPT(7,N,KK)=1E6*RHC	SUMP0069
OPT(8,N,KK)=1E6*RNOX	SUMP0070
GO TO MEAN,(50,70)	SUMP0071
C	SUMP0072
C STREAMTUBE AREA AND VELOCITY	SUMP0073
C	SUMP0074
50 AREA = 3.1415927*(RAD(N+1)**2-RAD(N)**2)/144.	SUMP0075
VEL=W(N)*SPVHAR/AREA	SUMP0076
OPT(1,N,1)=RAD(N)/12.	SUMP0077
OPT(2,N,1)=RAD(N+1)/12.	SUMP0078
OPT(3,N,1)=W(N)	SUMP0079
OPT(4,N,1)=VEL	SUMP0080
OPT(5,N,1)=F(1,N,1)	SUMP0081
OPT(6,N,1)=T(1)	SUMP0082
OPT(7,N,1)=F(2,N,1)	SUMP0083
OPT(8,N,1)=T(2)	SUMP0084
C	SUMP0085
C FUEL AND CONTAMINANT SUMMATIONS	SUMP0086
C	SUMP0087
FFLOW=FBAR*W(N)	SUMP0088
SUM(1)=SUM(1)+FFLOW	SUMP0089
SUM(2)=SUM(2)+FFLOW*EICD	SUMP0090
SUM(3)=SUM(3)+FFLOW*FIMC	SUMP0091
SUM(4)=SUM(4)+FFLOW*FINDX	SUMP0092
C	SUMP0093
C MEAN IMPACT PRESSURE	SUMP0094
C	SUMP0095
TAU=YY(V,1)*SPV(1)/SPV(2)/(1.0+YY(N,1)*(SPV(1)/SPV(2)-1.0))	SUMP0096
DO 60 IP=1,2	SUMP0097
FGAM(IP)=SQRT(GAM(IP)*(2.0/(GAM(IP)+1.0))**((GAM(IP)+1.0)	SUMP0098
1/(GAM(IP)-1.0)))	SUMP0099
60 W(IP)=VEL/SQRT(49723.928*GAM(IP)/MW(IP)*T(IP))	SUMP0100

	CTAU=1.0-TAU	SUMP0101
	PT=P0*(TAU**2*PRAT(M(1),GAM(1))+CTAU**2*PRAT(M(2),GAM(2)))	SUMP0102
	1/(TAU**2+CTAU**2)	SUMP0103
C		SUMP0104
C	GAS SAMPLE RICH FRACTION	SUMP0105
C		SUMP0106
	RW1W2=SQRT((T(2)+VEL**2/50075.458/CP(2)))	SUMP0107
	1/(T(1)+VEL**2/50075.458/CP(1))*MWT(1)/MWT(2))	SUMP0108
	RW1W2=RW1W2*FGAM(1)/FGAM(2)*PRAT(M(1),GAM(1))/PRAT(M(2),GAM(2))	SUMP0109
	YYS=TAU*RW1W2/(TAU*RW1W2+CTAU)	SUMP0110
C		SUMP0111
C	APPARENT MEAN TEMPERATURE (ASPIRATED THERMOCOUPLE)	SUMP0112
C		SUMP0113
	TTC=(TAUA*(T(1)/T(2))**0.45*RW1W2**0.6*T(1)+CTAU*T(2))	SUMP0114
	1/(TAUA*(T(1)/T(2))**0.45*RW1W2**0.6+CTAU)	SUMP0115
	OPT(9,N,1)=PT	SUMP0116
	OPT(10,N,1)=TTC	SUMP0117
C		SUMP0118
C	APPARENT MEAN PROPERTIES OF GAS SAMPLE	SUMP0119
C		SUMP0120
	ASSIGN 70 TO MEAN	SUMP0121
	KK=5	SUMP0122
	YYY=YYS	SUMP0123
	GO TO 30	SUMP0124
C		SUMP0125
	70 CONTINUE	SUMP0126
C		SUMP0127
C	OVERALL EMISSIONS INDICES	SUMP0128
C		SUMP0129
	EICD=1E3*SUM(2)/SUM(1)	SUMP0130
	FINC=1E3*SUM(3)/SUM(1)	SUMP0131
	FINDX=1E3*SUM(4)/SUM(1)	SUMP0132
C		SUMP0133
C	PRINT RESULTS	SUMP0134
C		SUMP0135
	CALL CLOCK(TIME,FLAPSD,MONTH,DAY)	SUMP0136
	DATA YEAR/75/	SUMP0137
	WRITE (6,1000) TIME,MONTH,DAY,YEAR,FLAPSD,TITLE,PRINT(LSTA),	SUMP0138
	((OPT(I,J,1),I=1,10),J=1,11)	SUMP0139
	WRITE(6,1001)	SUMP0140
	WRITE(6,1002)((OPT(I,J,2),I=1,10),J=1,11)	SUMP0141
	WRITE(6,1003)	SUMP0142
	WRITE(6,1002)((OPT(I,J,3),I=1,10),J=1,11)	SUMP0143
	WRITE(6,1004)(W(NT+1),SUM(1),EICD,FINC,FINDX,(SUM(I),I=2,4)	SUMP0144
C		SUMP0145
C	PUNCH CARDS FOR SUBSEQUENT PRINTING	SUMP0146
C	DATA FOR EACH TUBE - INNER, OUTER RADII, SAMPLE FUEL/AIR,	SUMP0147
C	EICD, FINC, FINDX, PPM CO, HC, NOX	SUMP0148
C		SUMP0149
	PUNCH 1005,TITLE,PRINT(LSTA),NT,((OPT(I,J,1),I=1,2),((OPT(I,J,3),	SUMP0150

	I=2,A),J=1,NT)	SUMP0151
C	RETURN	SUMP0152
C		SUMP0153
	1000 FORMAT(1H1,12X,40HANALYTICAL MODEL OF EXHAUST PLUMES FROM ,	SUMP0154
	'56HAFTERBURNING ENGINES TO PREDICT REACTION OF CONTAMINANT ,	SUMP0155
	'10HEMISSIONS./32H GENERAL ELECTRIC COMPANY ,	SUMP0156
	'21HAIRCRAFT ENGINE GROUP,9X,11HCOMPUTED AT,F6.2,10H HOURS ON ,	SUMP0157
	'2(T2,1H/1,12,7X,12HELAPSED TIME,F8.2,8H SECONDS/ 6X,20A6//	SUMP0158
	'34X,34HPRFILES AND CONTAMINANT RESIDUALS,F8.3,	SUMP0159
	'22H FEET FROM NOZZLE EXIT//9X,12HRADIUS, FEET,9X,13HGAS FLOW	SUMP0160
	'8HVELOCITY,11X,9HRICh PART,17X,9HLEAN PART,12X,10HPREDICTED ,	SUMP0161
	'10HINDICATION/11H INNER,8X,5HOUTER,9X,5HPPS,10X,5HPPS,7X,	SUMP0162
	'2(26HFUEL/GAS TEMP,DEGR) ,21H PT,PSIA TT,DEGR//	SUMP0163
	'11(1X,1P10E13,5//)	SUMP0164
	1001 FORMAT(28X,42HAVERAGE GAS COMPOSITION UNDER FREE STREAM ,	SUMP0165
	'10HCONDITIONS)	SUMP0166
	1002 FORMAT(51H0 MASS FRACT FUEL/AIR EMISSIONS INDICES, ,	SUMP0167
	'11HLR/KLR FUEL,17X,23HANALYZER READINGS, PPMV//13H RICH PART,	SUMP0168
	'6X,5HRATIO,2X,2HCO,11X,2HHO,7X,27HNOX AS NO2 CO(DRIFT) ,	SUMP0169
	'9HHC AS CHN,7X,3HNOX//11(1X,1P10E13,5//)	SUMP0170
	1003 FORMAT(28X,39HAVERAGE GAS COMPOSITION AS MODIFIED BY ,	SUMP0171
	'12HSAMPLE PROBE)	SUMP0172
	1004 FORMAT(28X,41HINTEGRATED PROFILE AVERAGES (FREE STREAM ,	SUMP0173
	'11HCONDITIONS)/23H0 TOTAL FLOWS, PPS,9X,17HEMISSIONS INDICES,	SUMP0174
	'13H, LB/KLR FUEL,13X,22HCONTAMINANT FLOWS, PPS/12H GAS MIX,	SUMP0175
	'7X,7HEUEL ,2(7X,2HCO,11X,2HHO,7X,10HNOX AS NO2)//1X,1P10E13.5)	SUMP0176
	1005 FORMAT(2(6X,10A6//),17X,2HX=F7.3,3H FT,17X,1H(,12,7H TUBES)/	SUMP0177
	'(1P6E12,4/36X,5E12,4))	SUMP0178
C		SUMP0179
	END	SUMP0180
		SUMP0181

CTABPRT TABLE PRINTOUT		
	SUBROUTINE TABPRT(NAME,A,NA,NCOL1)	TABPRT01
	DIMENSION A(10)	TABPRT02
C		TABPRT03
C	INPUT-	TABPRT04
C	NAME = ARRAY NAME TO BE PRINTED	TABPRT05
C	A = ARRAY TO BE PRINTED	TABPRT06
C	NA = NUMBER OF ELEMENTS	TABPRT07
C	NCOL1 = NUMBER OF COLUMNS TO BE USED IN THE PRINT FORMAT (MAXIMUM	TABPRT08
C	IITAB = LOCATION OF FIRST ELEMENT IN A-ARRAY TO BE PRINTED	TABPRT09
C		TABPRT10
	COMMON /CHITS / BITS,BLANK	TABPRT11
	COMMON /CTABPRT/ IITAB	TABPRT12
C		TABPRT13
	EQUIVALENCE (I8,B), (IC,C), (LSPACE,ASPACE)	TABPRT14
	DIMENSION FMT(12)	TABPRT15
	REAL I12	TABPRT16
C		TABPRT17
	DATA IBCT/00100000000000/	TABPRT18
	DATA ZERO/0./	TABPRT19
	DATA FMT(1)/67H(1X,15	TABPRT20
	1)/	TABPRT21
	DATA	TABPRT22
	A F1, F3, F6, E5, BCD, OCT, I12/	TABPRT23
	86H,F12.1, 6H,F12.3, 6H,F12.6, 6H,E12.4, 6H,6X,A6, 6H,8X,04, 4H,I12	TABPRT24
	C/	TABPRT25
C		TABPRT26
	NCOL = MIN0(NCOL1,10)	TABPRT27
	N8 = NA	TABPRT28
C		TABPRT29
	WRITE HEADING	TABPRT30
C	WRITE (6,1000) NAME	TABPRT31
		TABPRT32
C		TABPRT33
	45 I1 = IITAB	TABPRT34
	I = I1	TABPRT35
	I2 = 0	TABPRT36
C		TABPRT37
	WRITE LINE SPACE	TABPRT38
C	47 WRITE (6,1002)	TABPRT39
	LOCATION OF NEXT LINE SPACE IS GIVEN BY THE VALUE OF A(I+1)	TABPRT40
	ASPACE= A(I+1)	TABPRT41
	IF(LSPACE.LE.1 .OR. LSPACE.GE.IBC1) LSPACE=IBCI	TABPRT42
	LSPACE= LSPACE+I-1	TABPRT43
	GO TO 110	TABPRT44
C		TABPRT45
	BEGIN LOOP TO DEFINE LINE FORMAT	TABPRT46
C	48 I1 = 1	TABPRT47
		TABPRT48
C		TABPRT49
	CHECK FOR NORMALIZED FLOATING NUMBER	TABPRT50
	50 B = A(I)	

C	= B+ZERO	TABPR151
	IF(B,EQ,BITS) GO TO 85	TABPR152
	IF(IC,NE,IR) GO TO 80	TABPR153
	IF(B,EQ,0,) GO TO 65	TABPR154
	B = ADS(B)	TABPR155
	IF(B,LE,1,F=9 ,OR, B,GE,9,F+18) GO TO 80	TABPR156
C		TABPR157
C	REAL NUMBER	TABPR158
	FMT(II+1)=E5	TABPR159
	IF(B,LT,1,E=3 ,OR, B,GE,1,E8) GO TO 90	TABPR160
65	FMT(II+1)=F6	TABPR161
	IF(B,GE,1,E3) FMT(II+1)=F3	TABPR162
	IF(B,GE,1,E5) FMT(II+1)=F1	TABPR163
	GO TO 90	TABPR164
C		TABPR165
C	INTEGER AND BCD	TABPR166
80	FMT(II+1)=I12	TABPR167
	IF(IARS(ID),GT,IRCI) FMT(II+1)=BCD	TABPR168
	GO TO 90	TABPR169
C		TABPR170
C	OCTAL	TABPR171
85	FMT(II+1)=OCT	TABPR172
C		TABPR173
90	II = II+1	TABPR174
	I = I+1	TABPR175
	IF(I,GT,LSPACE) GO TO 100	TABPR176
	IF(II,LE,NCUL .AND. I,LE,NB) GO TO 50	TABPR177
100	I2 = I-1	TABPR178
	WRITE (6,FMT) II,(A(I),I=I1,I2)	TABPR179
	I1 = I2+1	TABPR180
110	IF(I2,GE,NB) GO TO 990	TABPR181
	IF(I,GT,LSPACE) GO TO 47	TABPR182
	GO TO 48	TABPR183
990	IITAB = 1	TABPR184
	RETURN	TABPR185
C		TABPR186
1000	FORMAT(/2X,A6)	TABPR187
1002	FORMAT(1H)	TABPR188
	END	TABPR189

CTDSEQ	TRIDIAGONAL MATRIX-SIMULTANEOUS EQUATIONS	TDSEQ001
C		TDSEQ002
	SUBROUTINE TDSEQ(A,N2,ID,FN)	TDSEQ003
C*		TDSEQ004
C*	A= COEFFICIENT ARRAY--A,B,C,D BY COLUMNS	TDSEQ005
C*	FIRST COLUMN DESTROYED BY SOLUTION VECTOR	TDSEQ006
C*****	SOLUTION VECTOR RETURNED AS FIRST COLUMN OF A ARRAY	TDSEQ007
C*		TDSEQ008
C*		TDSEQ009
C*	IF NORMAL DERIVATIVE CONDITIONS ARE ENFORCED USING 1ST TWO	TDSEQ010
C*	POINTS ABOVE OR BELOW THE BOUNDARIES, A(1) AND/OR A(3*N2),NE, 0,-	TDSEQ011
C*	THIS CONDITION IS FLAGGED BY IL AND/OR IU=1,	TDSEQ012
	DIMENSION A(800)	TDSEQ013
C*		TDSEQ014
	ENTRY TDSEQ1	TDSEQ015
	ISIZE=ID	TDSEQ016
	I2=ISIZE+1	TDSEQ017
	IF (A(I2)) 12,11,12	TDSEQ018
11	FN=1.	TDSEQ019
	GO TO 50	TDSEQ020
12	I3=2*ISIZE+1	TDSEQ021
	I4=3*ISIZE+1	TDSEQ022
	IU=0	TDSEQ023
	IL=0	TDSEQ024
	G=0,	TDSEQ025
	IF(A(1),NE,0.) IL=1	TDSEQ026
	INM=I3+I2-1	TDSEQ027
	IF(A(INM),NE,0.) IU=1	TDSEQ028
	G=A(1)/A(I2)	TDSEQ029
	A(I3)=A(I3)/A(I2)	TDSEQ030
	A(I4)=A(I4)/A(I2)	TDSEQ031
	N=N2	TDSEQ032
	DO 10 I=2,N	TDSEQ033
	I2=ISIZE+I	TDSEQ034
	I3=I2+ISIZE	TDSEQ035
	I4=I3+ISIZE	TDSEQ036
	IF(I,NE, N ,OR. IU,NE,1) GO TO 15	TDSEQ037
	A(I)=A(I)-A(I3)*A(I3-2)	TDSEQ038
	A(I4)=A(I4)-A(I3)*A(I-2)	TDSEQ039
15	CONTINUE	TDSEQ040
	A(I2)=A(I2)-A(I)*A(I3-1)	TDSEQ041
	IF (A(I2)) 13,11,13	TDSEQ042
13	IF(I,EQ,2 .AND. IL,EQ,1) A(I3)=A(I3)-A(I)*G	TDSEQ043
	IF(I,EQ,4 .AND. IU,EQ,1) GO TO 10	TDSEQ044
	A(I3)=A(I3)/A(I2)	TDSEQ045
10	A(I)=(A(I4)-A(I)*A(I-1))/A(I2)	TDSEQ046
	I=N	TDSEQ047
20	I=I-1	TDSEQ048
	IF (I) 40,40,30	TDSEQ049
30	I3=2*ISIZE+I	TDSEQ050

A(I)=A(I)-A(I3)*A(I+1)
IF(IL, EQ, 1, AND, I, EQ, 1) A(I)=A(I)-G*A(I+2)
GO TO 20
40 FN=0.
50 RETURN
END

TDSEQ051
TDSEQ052
TDSEQ053
TDSEQ054
TDSEQ055
TDSEQ056

CTFMH	CALCULATE TEMPERATURE FROM ENTHALPY	TFMH0001
	SUBROUTINE TFMH (NK)	TFMH0002
	LOGICAL ENTRY1	TFMH0003
	COMMON /INDATA/ N,HF,WAR,T2,BETA,T25,FAR5,FINO2C,P0,	TFMH0004
*	RCO(1),RCO2(1),RHC(1),RNOX(1),	TFMH0005
*	PT(1),PS(1),BLOC(1),QCO(1)	TFMH0006
REAL	N	TFMH0007
COMMON /JETDAY/	NPTS,RAD(12),TS(12),IU(12),SPV(12),MWT(12),	TFMH0008
*	CP(12),FUUL(12),SPALOG(12),TKE(12),OTHER1(36)	TFMH0009
REAL	MWT	TFMH0010
COMMON /GASCOMP/	RICH(12,2),FUEL(2,12,2),FNTH(2,12,2),	TFMH0011
*	CONC(16,2,12,2),HCINCP(12,2),OTHER(192)	TFMH0012
COMMON /GASTMW/	TG(2,12,2),MWTG(2,12,2),TAU(12,2),CPG(2,12,2)	TFMH0013
REAL	MWTG	TFMH0014
COMMON /DPCTRL/	TITLE(20),PRINT(30)	TFMH0015
COMMON /STCTRL/	LSTA,FINAL,CHEMK,FIRSTH,FIRSTC,XC,DXC,DUMST(10)	TFMH0016
LOGICAL	FINAL,FIRSTH,FIRSTC	TFMH0017
INTEGER	CHEMK	TFMH0018
COMMON /CPRINT/	PDUM(20)	TFMH0019
COMMON /GHSC	/ GRT(25),HRT(25),SR(25),CPR(25),DCPR(25)	TFMH0020
DATA	IG1,ENTRY1,R/3200.,T,1.98596/	TFMH0021
C		TFMH0022
C		TFMH0023
	DO 100 K=1,NK	TFMH0024
	DO 100 J=1,2	TFMH0025
	HGT = FNTH(J,K,2)	TFMH0026
	IF(.NOT. ENTRY1) GO TO 10	TFMH0027
	T1 = TG1	TFMH0028
	GO TO 15	TFMH0029
10	T1 = TG(J,K,2)	TFMH0030
15	DO 30 KI=1,30	TFMH0031
	CALL THRM(T1/1.8,1.)	TFMH0032
	CPG1 = 0.	TFMH0033
	HG1 = 0.	TFMH0034
	DO 20 I=1,16	TFMH0035
	CPG1 = CPG1+CONC(I,J,K,2)*CPR(I)	TFMH0036
	HG1 = HG1+CONC(I,J,K,2)*HRT(I)	TFMH0037
20	CONTINUE	TFMH0038
	HG1 = HG1*R*T1	TFMH0039
	DH = HGT-HG1	TFMH0040
	CPG1 = CPG1*R	TFMH0041
	DT = DH/CPG1	TFMH0042
	DTA = ABS(DT)	TFMH0043
	IF(DTA,LE. .01) GO TO 99	TFMH0044
	T1 = T1+DT	TFMH0045
	IF(T1,LE.189.) T1=189.	TFMH0046
30	CONTINUE	TFMH0047
	WRITE (6,35) T1,DT,HG1,DH	TFMH0048
35	FORMAT(/2X,19HTEMP. NOT CONVERGED/2X,4E16.8)	TFMH0049
99	TG(J,K,2)= T1	TFMH0050

CPG(J,K,2) = CPG1
100 CONTINUE
ENTRY1 = .FALSE.

C
200 RETURN
END

TFMH0051
TFMH0052
TFMH0053
TFMH0054
TFMH0055
TFMH0056

CTFMH1	CP+ T FROM H	TFMH1001
	SUBROUTINE TFMH1(C,H,T,CP)	TFMH1002
	COMMON /GMS / GRT(25),HRT(25),SR(25),CPR(25),DCPR(25)	TFMH1003
	DIMENSION C(1)	TFMH1004
	LOGICAL ENTRY1	TFMH1005
	DATA TG1,ENTRY1,R/3200.,T,1.98596/	TFMH1006
C		TFMH1007
	HGT = H	TFMH1008
	IF(.NOT. ENTRY1) GO TO 1	TFMH1009
	T1 = TG1	TFMH1010
1	DO 10 K1=1,30	TFMH1011
	CALL THRM(T1/1.8,1.)	TFMH1012
	CPG1 = 0.	TFMH1013
	HG1 = 0.	TFMH1014
	DO 5 I=1,16	TFMH1015
	CPG1 = CPG1+C(I)*CPR(I)	TFMH1016
	HG1 = HG1+C(I)*HRT(I)	TFMH1017
5	CONTINUE	TFMH1018
	HG1 = HG1*R*T1	TFMH1019
	DH = HGT-HG1	TFMH1020
	CPG1 = CPG1*R	TFMH1021
	DT = DH/CPG1	TFMH1022
	IF(ABS(DT).LT. 0.01) GO TO 15	TFMH1023
	T1 = T1+DT	TFMH1024
	IF(T1.LE.189.) T1=189.	TFMH1025
10	CONTINUE	TFMH1026
	WRITE (6,14) T1,DT,HG1,DH	TFMH1027
14	FORMAT(//2X,19HTEMP. NOT CONVERGED/2X,4E16.8//)	TFMH1028
15	T = T1	TFMH1029
	CP = CPG1	TFMH1030
	ENTRY1 = .FALSE.	TFMH1031
	RETURN	TFMH1032
	END	TFMH1033

CTHRUST	CALCULATE NET THRUST--LBP	THRUST01
SUBROUTINE THRUST(NX)		THRUST02
INTEGER TWOJ, ITWO		THRUST03
LOGICAL SUPC, SUPSTP		THRUST04
LOGICAL SURSON		THRUST05
LOGICAL TRUURL		THRUST06
LOGICAL DPRIN		THRUST07
LOGICAL LAST, CORSTP, ADDP, ENTRY1, IER		THRUST08
LOGICAL ENF, ERR		THRUST09
LOGICAL AXI, XPRN, CHPRS, QJET, TURBJ, CORE		THRUST10
REAL KCP, MUL, MUEFF, MACH		THRUST11
REAL MJET, ME, MUREP		THRUST12
COMMON /JETTWO/		THRUST13
* TWO, DIAO, MJETO, TJETO, VJL, J		THRUST14
* PTJETO, TIJETO, NJD		THRUST15
REAL MJETO, MACHO		THRUST16
COMMON /BCO/ UO, EO, TWO		THRUST17
COMMON /CTRL2/		THRUST18
* EDGE1, SFI, MERGE, XMERGE, YMERGE,		THRUST19
* SLOPE1, SLOPE2, CEPT1, CEPTO		THRUST20
COMMON /MERGET/ MER, MERSTP, XMRG		THRUST21
LOGICAL TWO, MERGE, MER, MERSTP		THRUST22
COMMON /SETNEW/ LEDGE, LCOEN		THRUST23
COMMON /INP1/ ENTRY1		THRUST24
COMMON /MISC/ PM(10)		THRUST25
COMMON /PARAM/		THRUST26
* AL(200), RE(200), OM(200),		THRUST27
* EPS(200), DL(200),		THRUST28
* VAR(200), DVAR(200),		THRUST29
* SM1(200), NM1, SM(200), NM		THRUST30
* DX,		THRUST31
* BI, CII, DI,		THRUST32
* AN, BN, DN		THRUST33
C*		THRUST34
C***** INPUT COMMON		THRUST35
C*		THRUST36
COMMON /INPJET/		THRUST37
* DIAJ, MJET, TJET, PTJET, VJET,		THRUST38
* TIJET,		THRUST39
* PE, VE, ME, TIE, TE,		THRUST40
* AXI, NJ, NMAX,		THRUST41
* XJ(100), XPRN(100),		THRUST42
* GAM, RG, PR, PRT,		THRUST43
* SC, TREP, MUREP		THRUST44
C*		THRUST45
C***** CONTROL COMMON		THRUST46
C*		THRUST47
COMMON /CTRL/		THRUST48
* NXTA, CHPRS, QJET, TURBJ, COEF(10),		THRUST49
* NPU, NPD, DXC, XU, XDD,		THRUST50

* DSTORE(800)	THRUST51
C*	THRUST52
C***** PROFILE COMMON	THRUST53
C*	THRUST54
COMMON /PRNF/ PSI(200),Y(200),UD(200),THD(200),ED(200)	THRUST55
C*	THRUST56
C***** CONSTANT AND ERROR COMMON	THRUST57
C*	THRUST58
COMMON /CNFRR/ BITS , ERR , GC , GCJ , FOOT	THRUST59
C*	THRUST60
C***** BOUNDARY CONDITION COMMON	THRUST61
C*	THRUST62
COMMON /BC/ UEDGE , EEDGE , THEDGE	THRUST63
C*	THRUST64
C***** POTENTIAL CORE COMMON	THRUST65
C*	THRUST66
COMMON /CORED/ XCORE , CORE , CORSTP	THRUST67
C*	THRUST68
COMMON /SUPER/ SUPC,SUPSTP,XSUP	THRUST69
C***** SCALER (UNITS CONVERSION) COMMON	THRUST70
C*	THRUST71
COMMON /SCALER/ SP , SY , SLEN	THRUST72
C*	THRUST73
COMMON /JET/	THRUST74
* B(100) , UC(100) , TC(100) , TIC(100) ,	THRUST75
* PTC(100) , WJ(100) , YJ(100)	THRUST76
C*	THRUST77
COMMON /PROJIT/	THRUST78
* P , PRL , PRTT , RGAS , SCC ,	THRUST79
* TREFF , VSREF , MACH , XLC ,	THRUST80
* REFI , C , CHI , RNORM ,	THRUST81
* RHO(200) , MUL(200) , KCP(200) ,	THRUST82
* MUEFF(200) , XLN(200) , DK(200) , RETURB(200)	THRUST83
COMMON /XPRIN/ DPRIN	THRUST84
COMMON /EDGE/ YJETE , SFEDGE	THRUST85
COMMON /UMFSH/ DUMUI(4),CXPC,EXTP,NRPD	THRUST86
C*	THRUST87
COMMON /MIXER/ MIX,RD(100),XD(100),CF,YR(100)	THRUST88
LOGICAL MIX	THRUST89
COMMON /GLOBAL/ MAXIT,SUPB,NIT,PSID,YDD,YDC,	THRUST90
* P1,P2,UCL,TOL,UPSTRM,CVG	THRUST91
LOGICAL SUPR,CVG,UPSTRM	THRUST92
COMMON /ACNVG/ YCD(100),PD(100),INDC(100), CHOKE, CHOKED	THRUST93
LOGICAL CHOKE, CHOKED	THRUST94
COMMON /DEIT/ CLSP(100)	THRUST95
COMMON /STA2/ MACH2,T92,S92,V2,RHO2,DPDX2	THRUST96
REAL MACH2	THRUST97
COMMON /BCMIX2/ GRADU,TH,MUH,RHOM,PTT,TTE	THRUST98
REAL MUH	THRUST99
COMMON /THERM/ GHC(200),CP(200)	THRUST00

C*	COMMON /OUTMIX/ NXORIG	THRUST01
	COMMON /CBODY/ YCB(100),CLSPCB(100),YCB1,UCL1	THRUST02
	COMMON /THRST/ WV(100)	THRUST03
	DIMENSION STORI(200),THR(200)	THRUST04
	EQUIVALENCE (STORI(1),AL(1)),(THR(1),BE(1))	THRUST05
	DATA PI/3.14159265/	THRUST06
C*		THRUST07
C*	IF MIXING ZONE HAS INTERSECTED PLUG-- TERMINATE	THRUST08
C*		THRUST09
	EPS=1.	THRUST10
	IF(AXI) EPS=2.	THRUST11
	1 YPLUG=YCB(NX)	THRUST12
	UP=YOF(YPLUG,Y,UD,1,NPD)	THRUST13
	IF(UP.EQ. UCL1) GO TO 10	THRUST14
	IF(YPLUG.EQ. 0.) GO TO 10	THRUST15
	WRITE (6,100) NX	THRUST16
	100 FORMAT(//6X,28H MIXING ZONE INTERSECTED PLUG/,6X,7HSTATION,	THRUST17
	* 17//12X,22H CALCULATION TERMINATED)	THRUST18
	FRR=TRUE.	THRUST19
	GO TO 1000	THRUST20
C*		THRUST21
C*	CALCULATE THRUST OF JET STREAM	THRUST22
C*		THRUST23
	10 DO 15 L=1,NPD	THRUST24
	REXP=1.	THRUST25
	IF(AXI) REXP=Y(L)	THRUST26
	15 STORI(L)=RHO(L)*UD(L)*UCL1*REXP	THRUST27
	THR(1)=0.	THRUST28
	CALL INIG(STORI,Y,THR,2,NPD)	THRUST29
	TJ=THR(NPD)/GC	THRUST30
C*		THRUST31
	IF(UPSTRM) GO TO 19	THRUST32
	TU=0.	THRUST33
	GO TO 50	THRUST34
C*		THRUST35
C*	THRUST OF UNFTRAINED FLOW	THRUST36
C*		THRUST37
	19 IF(NX.NE.1) GO TO 20	THRUST38
	TU=144.*PE/(RG*TE)*(VE/VJET)**2/(GC*EPS)*(YR(1)**EPS-Y(NPD)**EPS)	THRUST39
	GO TO 50	THRUST40
	20 TU=RHO2*(V2/VJET)**2/(GC*EPS)*(YDD**EPS-YJETE**EPS)	THRUST41
C*		THRUST42
C*		THRUST43
C*	CALCULATE NET THRUST	THRUST44
C*		THRUST45
	50 TERM=VJET**2*DIAJ/FOOT	THRUST46
	IF(AXI) TERM=PI*DIAJ*TERM/(2.*FOOT)	THRUST47
	TN=TERM*(TJ+TU)	THRUST48
C*		THRUST49
		THRUST50

60 WV(NX)TN
Ca
1000 RETURN
END

THRUST51
THRUST52
THRUST53
THRUST54

CTHRMM	CALCULATES (DIMENSIONLESS) THERMODYNAMIC PROPERTIES	THRMM001
C	FROM POLYNOMIAL CURVE FITS	THRMM002
	SUBROUTINE THRM (T,HONLY)	THRMM003
C		THRMM004
C	THIS ROUTINE CALCULATES (DIMENSIONLESS) THERMODYNAMIC PROPERTIES	THRMM005
C	FROM POLYNOMIAL CURVE FITS	THRMM006
C		THRMM007
	LOGICAL NEXT	THRMM008
C		THRMM009
	COMMON/COND/DUM(33),LS,LSP3,NEXT	THRMM010
	COMMON/GHSC/GRT(25),HRT(25),SR(25),CPR(25),OCPR(25)	THRMM011
	COMMON/TCDF/C(7,2,25),TLOW,TMID,THI	THRMM012
C		THRMM013
	$F(T) = A1 + T * (A2 + T * (A3 + T * (A4 + T * A5)))$	THRMM014
C		THRMM015
	IF (T .EQ. TPREV) RETURN	THRMM016
C		THRMM017
	IF (0.35*TLOW .LE. T .AND. T .LE. THI) GO TO 3	THRMM018
	IF (T .LE. 1.20*THI) GO TO 2	THRMM019
C		THRMM020
	WRITE (6,100) T	THRMM021
100	FORMAT (7H0(THRM),5X,5HERROR,3X,3HT =,F8.2,16H IS OUT OF RANGE)	THRMM022
	NEXT = .TRUE.	THRMM023
	RETURN	THRMM024
C		THRMM025
2	WRITE (6,101) T	THRMM026
101	FORMAT (7H0(THRM),5X,7HWARNING,3X,3HT =,F8.2,16H IS OUT OF RANGE,	THRMM027
	* 4X,2BHEXTRAPOLATED VALUES RETURNED)	THRMM028
C		THRMM029
	LOCATE PROPER TEMPERATURE RANGE	THRMM030
3	K = 2	THRMM031
	IF (T .GT. TMID) K = 1	THRMM032
C		THRMM033
	DO 4 I=1,LS	THRMM034
C		THRMM035
	COMPUTE H/(R*T)	THRMM036
	A1 = C(1,K,I) + C(6,K,I)/T	THRMM037
	A2 = C(2,K,I)/2.	THRMM038
	A3 = C(3,K,I)/3.	THRMM039
	A4 = C(4,K,I)/4.	THRMM040
	A5 = C(5,K,I)/5.	THRMM041
4	HRT(I) = F(T)	THRMM042
	IF (HONLY .EQ. 0.) RETURN	THRMM043
C		THRMM044
	TPREV = T	THRMM045
	DO 5 I=1,LS	THRMM046
C		THRMM047
	COMPUTE G/(R*T)	THRMM048
	A1 = C(1,K,I)*(1.-ALOG(T)) + C(6,K,I)/T - C(7,K,I)	THRMM049
	A2 = -C(2,K,I)/2.	THRMM050

A3 = -C(3,K,I)/6.

A4 = -C(4,K,I)/12.

A5 = -C(5,K,I)/20.

GRT(I) = F(T)

C

C

COMPUTE S/R

A1 = C(1,K,I)*ALOG(T) + C(7,K,I)

A2 = C(2,K,I)

A3 = C(3,K,I)/2.

A4 = C(4,K,I)/3.

A5 = C(5,K,I)/4.

SR(I) = F(T)

C

C

COMPUTE CP/R

A1 = C(1,K,I)

A2 = C(2,K,I)

A3 = C(3,K,I)

A4 = C(4,K,I)

A5 = C(5,K,I)

CPR(I) = F(T)

C

C

COMPUTE (DCP/DI)/R

A1 = C(2,K,I)

A2 = 2.*C(3,K,I)

A3 = 3.*C(4,K,I)

A4 = 4.*C(5,K,I)

A5 = 0.

5 DCPR(I) = F(T)

C

RETURN

END

THRMM051

THRMM052

THRMM053

THRMM054

THRMM055

THRMM056

THRMM057

THRMM058

THRMM059

THRMM060

THRMM061

THRMM062

THRMM063

THRMM064

THRMM065

THRMM066

THRMM067

THRMM068

THRMM069

THRMM070

THRMM071

THRMM072

THRMM073

THRMM074

THRMM075

THRMM076

THRMM077

THRMM078

THRMM079

THRMM080

THRMM081

CTRIDIA	SOLVES TRI-DIAGONAL SYSTEM OF EQUATIONS	TRIDIA01
C		TRIDIA02
	SUBROUTINE TRIDIA (J,NN)	TRIDIA03
C		TRIDIA04
C*	THIS ROUTINE FINDS THE SOLUTION OF A TRIDIAGONAL SYSTEM OF EQUATIONS	TRIDIA05
C*	THE MATRIX -A- CONTAINS THE TRIDIAGONAL COEFFICIENT MATRIX	TRIDIA06
C*	THE MATRIX -B- CONTAINS THE RHS VECTOR	TRIDIA07
C*	THE SIZE OF THE MATRIX IS (NN+1)X(NN+1)	TRIDIA08
C		TRIDIA09
	COMMON /CTRIDIA/ A(50,3),B(50)	TRIDIA10
	A(J,3)=A(J,3)/A(J,2)	TRIDIA11
	B(J)=B(J)/A(J,2)	TRIDIA12
	JP1=J+1	TRIDIA13
	NNM1=NN-1	TRIDIA14
	DO 20 N=JP1,NNM1	TRIDIA15
	A(N,2)=1./(A(N,2)-A(N,1)*A(N-1,3))	TRIDIA16
	A(N,3)=A(N,3)+A(N,2)	TRIDIA17
	20 B(N)=(B(N)-A(N,1)*B(N-1))*A(N,2)	TRIDIA18
C*	BACK SUBSTITUTION	TRIDIA19
C*	STORE SOLUTION VECTOR IN RHS VECTOR LOCATION	TRIDIA20
	N=NN	TRIDIA21
	A(N,2)=1./(A(N,2)-A(N,1)*A(N-1,3))	TRIDIA22
	B(N)=(B(N)-A(N,1)*B(N-1))*A(N,2)	TRIDIA23
260	IF (N=J) 300,300,270	TRIDIA24
270	N=N-1	TRIDIA25
	B(N)=B(N)-A(N,3)*B(N+1)	TRIDIA26
	GO TO 260	TRIDIA27
300	CONTINUE	TRIDIA28
	RETURN	TRIDIA29
	END	TRIDIA30

CXSIZE	X-STEP SIZE CONTROL	XSIZE001
	SUBROUTINE XSIZE (DX,X,REFL,NREG,LAST)	XSIZE002
	LOGICAL LAST	XSIZE003
	COMMON /SETNEW/ LKK,LC	XSIZE004
	COMMON /UMFSH/ DUMX(4),CXPC,CXTP,NREG	XSIZE005
	COMMON /CTRL/ DUMSX(9),C6,DUMSX1(N),XD,DUMSX2(800)	XSIZE006
C*		XSIZE007
C*	NREG=1, DX PROPORTIONAL TO B	XSIZE008
C*	NREG=2,3 DX PROPORTIONAL TO JET RADIUS	XSIZE009
C*		XSIZE010
C*		XSIZE011
	LAST=.FALSE.	XSIZE012
	DXT=XD-X	XSIZE013
	1 GO TO (10,20,20),NREG	XSIZE014
C*		XSIZE015
	10 DX=CXPC*REFL/C6	XSIZE016
	IF(DX.GE. .99*DXT) DX=DXT	XSIZE017
	XT=X+DX	XSIZE018
	GO TO 50	XSIZE019
C*		XSIZE020
	20 IF(LC.GT.10) GO TO 30	XSIZE021
	DX=.1*FLOAT(LC)*CXTP*REFL/C6	XSIZE022
	LC=LC+1	XSIZE023
	GO TO 40	XSIZE024
	30 DX=CXTP*REFL/C6	XSIZE025
	40 IF(DX.GE. .99*DXT) DX=DXT	XSIZE026
	XT=X+DX	XSIZE027
C*		XSIZE028
C*		XSIZE029
C*	CHECK FOR X.GI. CALCULATION STATION	XSIZE030
C*		XSIZE031
	50 IF(XT.LT.XQ) GO TO 100	XSIZE032
	LAST=.TRUE.	XSIZE033
	DX=XD-X	XSIZE034
C*		XSIZE035
	100 RETURN	XSIZE036
	END	XSIZE037

CYOFX11

FUNCTION YOF(X1, X, Y, IA, IB)

COMMON /YOFX1/ I

DIMENSION X(10), Y(10)

I1=IA+1

I2=IB

I=MINO(MAXO(I1, I), I2)

N=I2

40 IF(N) 42, 999, 42

42 N=N-1

53 F=(X1-X(I-1))/(X(I)-X(I-1))

55 IF(F) 60, 100, 70

60 IF(I-I1) 65, 100, 65

65 I=I-1

GO TO 40

70 IF(F-1.) 100, 100, 72

72 IF(I-I2) 74, 100, 74

74 I=I+1

GO TO 40

999 CALL ERROR

RETURN

100 YOF = Y(I-1) + (Y(I)-Y(I-1))*F

RETURN

END

YOFX1101

YOFX1102

YOFX1103

YOFX1104

YOFX1105

YOFX1106

YOFX1107

YOFX1108

YOFX1109

YOFX1110

YOFX1111

YOFX1112

YOFX1113

YOFX1114

YOFX1115

YOFX1116

YOFX1117

YOFX1118

YOFX1119

YOFX1120

YOFX1121

YOFX1122

YOFX1123

YOFX1124

SECTION 5.0

REFERENCES

1. Lyon, T.F., et. al., Development of Emissions Measurement Techniques for Afterburning Turbine Engines, AFAPL-TR-75-52, Air Force Aero Propulsion Laboratory, September 1975.
2. Heck, P.H., and Ferguson, D.R., Analytical Solution For Free Turbulent Mixing in Compressible Flows, AIAA Paper No. 71-4, January 1971.
3. Bittker, D.A., and Scullin, V.J., General Chemical Kinetics Computer Program For Static and Flow Reactions, With Application to Combustion and Shock-Tube Kinetics, NASA TN D-6586, January 1972.
4. Colley, W.C., Ferguson, D.R., Harsha, P.T., and Kenworthy, M.J., Scientific Technology of Hypersonic Ramjet Combustion Systems, AFAPL-TR-67-156, December 1967.
5. Spalding, D.B., "Concentration Fluctuations in a Round Turbulent Free Jet", Chemical Engineering Science, Volume 26, pp. 95-107, 1971.
6. Spalding, D.B., "Mathematical Models of Continuous Combustion", pp. 3-16 of Emissions from Continuous Combustion Systems, Cornelius, W. and Agnew, W.G. (Eds.), Plenum Press. 1972.
7. Lauder, B.E., and Spalding, D.C., Mathematical Models of Turbulence, Academic Press, 1972.
8. Anon., FORTRAN IV, Honeywell Information Systems, Pub. BN88.

ATE
LMED
-7